UNIVERSITÉ DE MONTRÉAL

UNE MÉTHODE ADAPTATIVE POUR L'APPROXIMATION DE FONCTIONS CONCAVES CROISSANTES

JEAN GUÉRIN
DÉPARTEMENT DE MATHÉMATIQUES
ET DE GÉNIE INDUSTRIEL
ÉCOLE POLYTECHNIQUE DE MONTRÉAL

MÉMOIRE PRÉSENTÉ EN VUE DE L'OBTENTION
DU DIPLÔME DE MAÎTRISE ÈS SCIENCES APPLIQUÉES
(MATHÉMATIQUES APPLIQUÉES)
MAI 2000

UNIVERSITÉ DE MONTRÉAL

ÉCOLE POLYTECHNIQUE DE MONTRÉAL

Ce mémoire intitulé :

UNE MÉTHODE ADAPTATIVE POUR L'APPROXIMATION DE FONCTIONS CONCAVES CROISSANTES

présenté par : GUÉRIN, Jean
en vue de l'obtention du diplôme de : Maîtrise ès sciences appliquées
a été dûment accepté par le jury d'examen constitué de :

M. SMITH, Benjamin T. , Ph.D., président
M. SAVARD, Gilles, Ph.D., membre et directeur de recherche
M. MARCOTTE, Patrice, Ph.D., membre et codirecteur de recherche
M. GAUVIN, Jacques, Ph.D., membre

REMERCIEMENTS

Je tiens tout d'abord à remercier pour leurs conseils et leur soutien mes deux codirecteurs, Gilles Savard et Patrice Marcotte, sans qui ce mémoire n'aurait pas vu le jour.

Je remercie également les membres du GERAD pour l'aide qu'ils m'ont apportée dans la réalisation de ce travail. En particulier, merci à Pascal Labit pour m'avoir aidé avec tant de pépins informatiques, et à Alexis Guigue pour des discussions aussi intéressantes que fructueuses.

RÉSUMÉ

Dans ce mémoire, nous considérons le problème d'approximation d'une fonction concave croissante par une fonction linéaire par morceaux construite à partir de l'évaluation de la fonction donnée en un nombre de points prédéterminé. Plus précisément, nous étudions la question suivante : étant donné un nombre fixé de points d'évaluation à choisir séquentiellement de gauche à droite de l'intervalle de définition d'une fonction concave croissante dont on peut en chaque point calculer la valeur ainsi qu'un surgradient, où doit-on placer ces points de façon à minimiser l'erreur, mesurée au sens intégral, dans le pire cas ?

L'approche que nous proposons pour répondre à cette question est celle de la programmation dynamique. Nous formulons d'abord le problème sous la forme d'une équation de récurrence, puis nous en donnons une solution analytique. La formule que nous obtenons, principal résultat de ce mémoire, donne la valeur de la plus petite erreur maximale dans le pire cas, sous les hypothèses considérées, ainsi que le premier point du processus séquentiel d'évaluation par lequel l'approximation est construite. Ces résultats s'appliquent également à d'autres classes de fonctions, par exemple les fonctions convexes croissantes.

De cette formule, nous tirons une procédure itérative pour l'approximation de fonctions de la classe considérée. Cette procédure est adaptative, c'est-à-dire qu'elle utilise l'information recueillie lors des évaluations précédentes pour déterminer le point suivant. Nous proposons également une méthode heuristique, un «algorithme du sandwich» optimal, pour l'approximation de telles fonctions. L'efficacité de ces méthodes est évaluée par des expériences numériques, en comparant les résultats obtenus avec

ceux provenant d'une approximation par des points uniformément distribués.

Enfin, nous décrivons une application des méthodes au problème d'équilibre bicritère sur un réseau. Celui-ci peut être résolu au moyen d'un algorithme de Frank et Wolfe généralisé dans lequel apparaît à chaque itération un sous-problème de plus court chemin paramétrique. La solution de ce dernier est donnée par une fonction linéaire par morceaux concave croissante qu'il est coûteux de calculer exactement. Les méthodes développées dans le présent travail peuvent être utilisées avantageusement pour en donner une approximation et ainsi accélérer la résolution du problème d'équilibre bicritère.

ABSTRACT

In this thesis we consider the problem of approximating a nondecreasing concave function using a piecewise linear function constructed by evaluating the given function at a predetermined number of points. Precisely, we study the following question : given a fixed number of evaluation points which are to be chosen sequentially from left to right on the interval of definition of a nondecreasing concave function, where should these points be located in order to minimize the error, measured in the integral sense, in the worst case?

Our approach to this problem is that of dynamic programming. We first formulate the problem in the form of a recurrence relation, then we find an analytic solution for this relation. The formula that we find, which is the main result of this thesis, gives the value of the worst case error, as well as the first point of the sequential evaluation process used to construct the approximation. These results also apply to other classes of functions, for example nondecreasing convex functions.

From this formula we obtain an iterative procedure for the approximation of functions in the class that we are considering. This procedure is adaptive, which means that it uses the information from previous evaluations to determine the next evaluation point. We also give a heuristic method, an optimal "sandwich algorithm", for the approximation of nondecreasing concave functions. The efficiency of these methods is evaluated by comparing our results with those obtained from an approximation with uniformly distributed points.

Finally, we describe an application of these methods to the bicriteria equilibrium problem. This problem can be solved using a generalized Frank and Wolfe algorithm

in which there is a parametric shortest path problem. The solution of the latter is given by a nondecreasing concave function that is costly to compute exactly. The methods developed is this thesis can be used to approximate its solution and thereby accelerate the solution of the bicriteria equilibrium problem.

TABLE DES MATIÈRES

LISTE DES TABLEAUX

LISTE DES FIGURES

CHAPITRE 1

INTRODUCTION

1.1 Présentation du problème

Soit f une fonction inconnue mais que l'on peut évaluer en tout point d'un intervalle donné. On cherche à approximer f par une fonction linéaire par morceaux construite à partir de l'information recueillie lors de l'évaluation de f en un certain nombre de points de l'intervalle. On veut construire une approximation qui minimise l'erreur commise en utilisant la valeur approchée, la façon de mesurer cette erreur ayant été préalablement définie. Si l'on suppose que l'évaluation de f est coûteuse en termes de calculs, on voudra utiliser le moins de points possibles pour atteindre une précision donnée, ce qui implique que le choix de ces points doit être fait de façon judicieuse. Le but du présent travail est d'élaborer une méthode permettant de faire un tel choix pour une certaine classe de fonctions.

Plus précisément, nous étudions des fonctions monotones continues, lisses par morceaux et dont la dérivée, lorsqu'elle existe, est aussi monotone. Les fonctions concaves croissantes font partie de cette classe et c'est en fait pour ces dernières que nous développons une procédure d'approximation. La généralisation aux autres fonctions de la classe considérée (par exemple les fonctions convexes décroissantes) est immédiate et est discutée à la section 2.7. En plus de pouvoir évaluer la fonction f en tout point d'un intervalle, nous supposerons que sa dérivée (ou un surgradient

si la dérivée n'est pas définie) peut aussi être évaluée et fait partie de l'information dont nous disposons sur f. Enfin, la mesure que nous avons choisie pour l'erreur est l'intégrale de la différence entre l'approximation et la fonction f, c'est-à-dire que l'erreur est mesurée en utilisant la norme $\mathcal{L}^1$.

De dire, comme plus haut, que la fonction à approximer est inconnue signifie que la procédure d'approximation sera élaborée en termes de la classe donnée et non pas d'une fonction particulière. Cette procédure devra donc s'appliquer à toutes les fonctions et en particulier à celle représentant le «pire cas» c'est-à-dire celle qui maximise l'erreur. On voudra donc une procédure qui minimise ce pire cas, et cette idée est au cœur des développements ultérieurs de ce travail. Il est à noter cependant que ce n'est pas la seule façon d'envisager la minimisation de l'erreur, comme nous le mentionnons à la section 1.3.1, où nous décrivons le problème d'approximation de manière plus formelle.

1.2 Deux exemples d'application

Nous présentons dans cette section deux problèmes où les fonction concaves croissantes jouent un rôle essentiel. Le premier, l'équilibre bicritère sur un réseau, est la motivation principale de ce travail et est exposé plus en détail. Le deuxième, la différenciation des produits, est une application en économie.

1.2.1 Équilibre bicritère

Cette première application porte sur l'affectation de trafic sur un réseau de transport. La solution de ce problème permet de déterminer les flots de véhicules sur un

réseau, c'est-à-dire la répartition des usagers sur les différents chemins reliant une origine à une destination. Une hypothèse généralement admise sur le comportement des usagers stipule que ceux-ci agissent de façon égoïste et que chaque usager cherche à minimiser son propre coût de transport. Ici «coût» peut désigner un coût monétaire, le temps, etc. Ceci donne lieu à un équilibre sur le réseau, où le coût de transport est le même pour tous les usagers. Ce principe, appelé *premier principe de Wardrop* constitue la base des développements de cette section.

Pour illustrer la notion d'équilibre, considérons un exemple simple mais célèbre, le «paradoxe de Braess» (voir [3]). Ici le réseau est constitué de quatre sommets et cinq arcs (figure 1.1). Les coûts pour chaque arc a dépendent du flot x_a sur cet arc et sont indiqués sur la figure. La demande est de six unités du sommet 1 au sommet 4. Il y a trois chemins reliant ces sommets, soit 1-2-4, 1-3-4 et 1-2-3-4. La solution

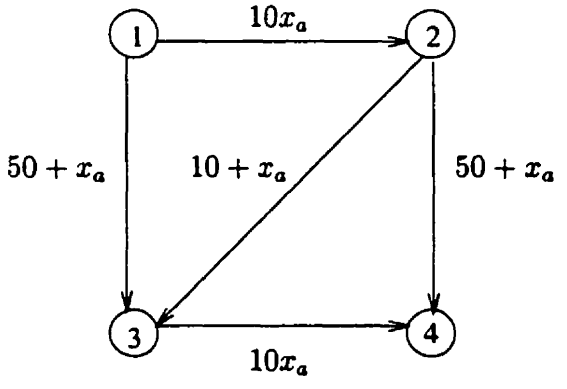


Figure 1.1 – Réseau de l'exemple

de ce problème est simple et s'obtient en résolvant le système d'équations obtenu en égalant les coûts sur les chemins (voir [22]). À l'équilibre il y a deux unités de flots sur chaque chemin le coût de transport (commun) est de 92. La raison pour laquelle cet exemple est qualifié de paradoxe est que si l'on interdit l'arc 2-3, on calcule qu'à l'équilibre il y a trois unités de flot sur chacun des deux chemins restants, pour un

coût de 83, ce qui est inférieur à ce qui a été trouvé précédemment. Ceci montre qu'à l'équilibre le coût individuel n'est pas nécessairement minimisé.

Dans cet exemple, le coût d'un chemin ne dépend que du flot sur les arcs le composant. Tous les usagers choisissent un chemin en fonction d'un même critère. Pour tenir compte des différences entre les usagers dans le calcul du coût, le modèle bicritère propose d'associer à chaque arc un coût de la forme

$$C_a(x_a, \alpha) = F_a(x_a) + \alpha G_a.$$

Ici F_a est une fonction du flot sur a et G_a est une constante. Par exemple, F_a pourrait être le temps de parcours de l'arc a et G_a un coût fixe pour son utilisation. Le paramètre α permet de varier le poids relatif du «temps» et de «l'argent» et est une caractéristique d'un usager, ou plus généralement d'une classe d'usagers. Ceux pour lesquels α est élevé accordent beaucoup d'importance au coût monétaire par rapport au temps de parcours, tandis que ceux pour qui α est faible au contraire considèrent que le temps est l'élément dominant dans le calcul du coût d'un chemin. Le paramètre α effectue une conversion entre «temps» et «argent» et est appelé *valeur du temps*. La distribution des valeurs de α parmi l'ensemble des usagers est donnée par une fonction de densité h, c'est-à-dire une fonction telle que

$$\int_0^{\alpha_{max}} h(\alpha)\, d\alpha = 1.$$

Nous décrivons maintenant de façon précise le problème d'équilibre bicritère. Puisque la solution complète de ce problème n'est pas l'objet de ce travail, nous nous contenterons d'en donner une formulation mettant en évidence les aspects qui nous intéressent plus particulièrement. Le lecteur trouvera dans l'article [12] de Marcotte plusieurs reformulations du problème d'équilibre bicritère, qui sont étudiées du point de vue théorique. Du point de vue pratique, l'auteur suggère pour la solution

numérique une généralisation de la méthode de Frank et Wolfe. Dans [13] Marcotte et Zhu étudient cette généralisation et prouvent sa convergence. Dans [15] et [22] on décrit une implantation de cette méthode et on présente les résultats d'expériences numériques.

Nous suivons dans ce qui suit le développement de [12]. Étant donné un graphe, on définit

α : la valeur du temps, un paramètre représentant une classe d'usagers.

h : la fonction de densité donnant la distribution de α dans la population. Cette fonction satisfait $\int_0^{\alpha_{max}} h(\alpha)\, d\alpha = 1$.

$x(\alpha)$: le vecteur des flots sur les chemins associé à la valeur α.

$\bar{x}$: le vecteur de flot total sur les chemins. On a

$$\bar{x} = \int_0^{\alpha_{max}} x(\alpha)\, d\alpha.$$

$\bar{X}$: l'ensemble des vecteurs de flots totaux sur les chemins qui sont réalisables, c'est-à-dire qui satisfont la demande ainsi qu'à la condition de conservation de flot.

$X(\alpha)$: l'ensemble des vecteurs de flots sur les chemins pour la classe α qui sont réalisables.

On supposera que $X(\alpha) = h(\alpha)\bar{X}$ pour tout α, autrement dit que la distribution des classes est la même pour toutes les paires origine-destination.

Le coût sur les chemins est donné par

$$C(\bar{x}) = F(\bar{x}) + \alpha G$$

où F est le vecteur des fonctions de délai sur les chemins et G le vecteur des coûts fixes. On remarque que le coût sur les chemins ne dépend que du flot total $\bar{x}$. Avec

ces notations l'équilibre s'exprime sous la forme d'une inégalité variationnelle :

$$x(\alpha) \in X(\alpha) \quad \forall \alpha$$

$$\bar{x} = \int_0^{\alpha_{max}} x(\alpha)\, d\alpha$$

$$\langle F(\bar{x}) + \alpha G, x(\alpha) - y(\alpha) \rangle \leq 0, \ \forall y(\alpha) \in X(\alpha), \ \forall \alpha. \tag{1.1}$$

La dernière inéquation traduit le fait que, pour un flot total fixé $\bar{x}$, le vecteur $x(\alpha)$ est à l'équilibre par rapport au coût $F(\bar{x}) + \alpha G$. En effet, l'inégalité implique qu'il n'est pas possible de trouver un vecteur $y(\alpha)$ réalisable qui diminuerait le coût.

L'inéquation variationnelle 1.1 est équivalente à

$$x(\alpha) \in \underset{y(\alpha) \in X(\alpha)}{\operatorname{argmin}} \langle F(\bar{x}) + \alpha G, y(\alpha) \rangle \quad \forall \alpha. \tag{1.2}$$

Puisque $X(\alpha) = h(\alpha)\bar{X}$, la solution de ce problème est de la forme $y(\alpha) = h(\alpha)\bar{y}$, donc 1.2 est équivalent à

$$\min_{\bar{y} \in \bar{X}} \langle F(\bar{x}) + \alpha G, \bar{y} \rangle. \tag{1.3}$$

Le flot total $\bar{x}$ étant fixé, ceci est un programme linéaire pour chaque α. On a donc affaire à un programme linéaire paramétrique. Plus précisément, il s'agit d'un problème de plus court chemin paramétrique (*problème PCCP*). Sa solution est de la forme

$$y(\alpha) = h(\alpha)\bar{y}^i, \ \text{ si } \alpha \in [\alpha_{i-1}, \alpha_i], \ \ i = 1, \ldots, N$$

où $\bar{y}^i$ est un sommet du polyèdre réalisable $\bar{X}$ et $0 \leq \alpha_1 \leq \ldots \leq \alpha_N$. Les α_i sont les *points de brisure* de la fonction objectif et correspondent à un changement de la base

optimale du programme linéaire 1.3 : la solution $\bar{y}^i$ est optimale sur $[\alpha_{i-1}, \alpha_i]$ mais cesse de l'être lorsque le paramètre α excède la valeur α_i et l'optimum est alors atteint en $\bar{y}^{i+1}$. Ceci est illustré à la figure 1.2, où $F^i + \alpha G^i = \langle F(\bar{x}) + \alpha G, \bar{y}^i \rangle$. Un programme linéaire paramétrique peut être résolu exactement en temps fini (voir [5] pour le cas général et [22] pour le PCCP). Pour déterminer la solution d'équilibre on se servira

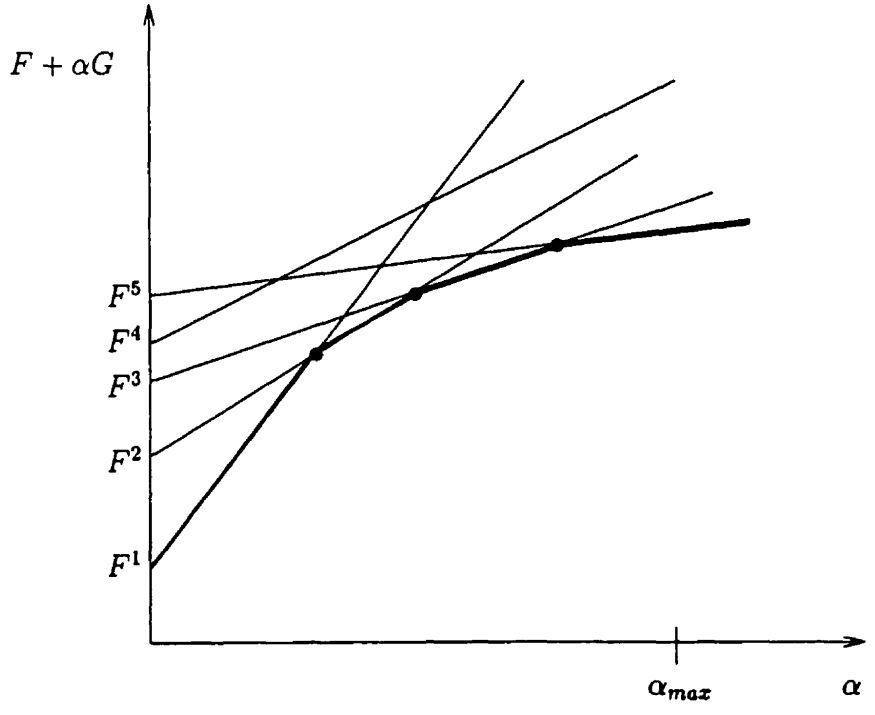


Figure 1.2 – Solution du PCCP

de l'information donnée par la solution du problème de PCCP. Celui-ci apparaît comme un sous-problème dans la méthode de Frank et Wolfe généralisée proposée par Marcotte, Nguyen et Tanguay dans [15] et [22]. Les auteurs y remarquent que le PCCP est la partie la plus coûteuse de cet algorithme. On veut donc remplacer la résolution exacte du sous-problème linéaire par une approximation dans le but de réduire le temps de calcul. Les résultats numériques dans [15] et [22] montrent que la perte de précision est compensée par la rapidité de la résolution. Dans [12] Marcotte suggère

d'approximer la fonction objectif du problème linéaire paramétrique en l'évaluant, c'est-à-dire en résolvant le problème de plus court chemin, en un petit nombre de points α_i adéquatement choisis. C'est ici que s'applique la méthode d'approximation que nous développons dans ce travail.

1.2.2 Différenciation des produits

Nous exposons dans cette section un exemple d'application à la science économique, tiré de [10]. Considérons un marché où plusieurs produits sont disponibles. Ces produits diffèrent entre eux par leur qualité, que l'on suppose quantifiable, et on associe à chacun une valeur de qualité q_i. Les produits diffèrent aussi quant à leur prix p_i, un produit de meilleure qualité ayant un prix plus élevé. La différence entre deux produits peut être mesurée relativement à un produit idéal, de qualité q^*, qui serait préféré à tous les autres par tous les consommateurs (voir figure 1.3).

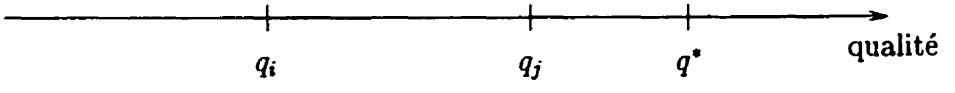


Figure 1.3 – Le produit q^*

Définissons le *coût de disparité* (*mismatch cost*), $d_i = q^* - q_i$, qui est la différence entre la qualité d'un produit donné et celle du produit idéal. Le produit j de meilleure qualité sera préféré au produit i si la différence de qualité compense l'augmentation du prix, c'est-à-dire si

$$p_j - p_i < d_i - d_j.$$

Supposons maintenant que les consommateurs ont des perceptions différentes de la différence de qualité. Ceci peut être modélisé au moyen d'un *paramètre de disparité*

α dont la distribution dans la population est connue. Le produit j sera préféré au produit i par les consommateurs ayant un paramètre de disparité α si

$$p_j - p_i < \alpha(d_i - d_j)$$

ou encore

$$p_j + \alpha d_j < p_i + \alpha d_i.$$

On reconnaît ici une situation similaire à l'exemple de la section précédente, le PCCP. On a un ensemble de droites $p_i + \alpha d_i$ dont le minimum $z(\alpha)$, pour chaque valeur de α, donne le coût pour un consommateur de la classe α (voir figure 1.4). Connaissant l'indice i de la droite correspondante, on connaît le produit choisi par le consommateur. Si

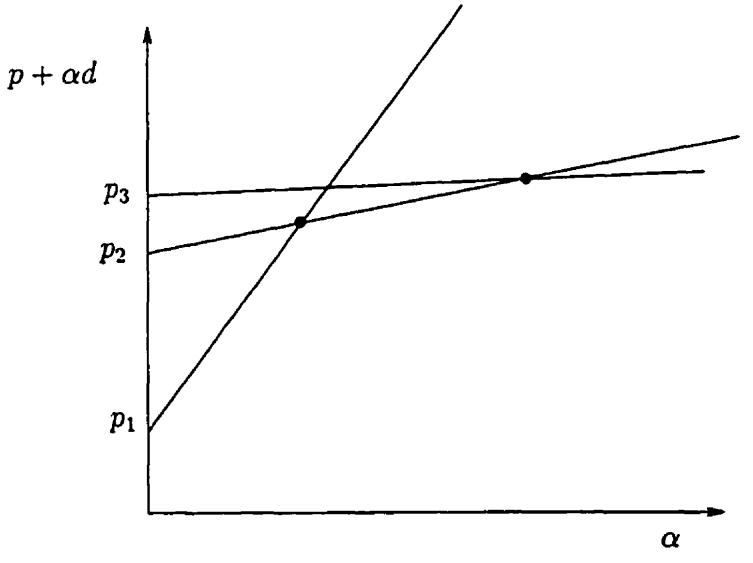


Figure 1.4 – Coût des produits

l'on peut déterminer tous les points de brisure $\bar{\alpha}_k$ de la fonction $z(\alpha) = \min_i\{p_i + \alpha d_i\}$, on connaît alors le choix de produit pour les consommateurs de chacun des intervalles $[\bar{\alpha}_{k-1}, \bar{\alpha}_k]$. Ceci peut être utile pour étudier le marché pour un produit, par exemple

pour déterminer si un produit offert à un certain prix sera intéressant pour les consommateurs. Sur la figure 1.5 (i), le produit 4 ne sera jamais choisi tandis qu'il le sera en (ii) par les consommateurs de classe $\alpha \in [\bar{\alpha}_1, \bar{\alpha}_2]$, lorsque son prix, p_4, diminue. Si le nombre de produits est assez grand alors le nombre de points de brisure peut

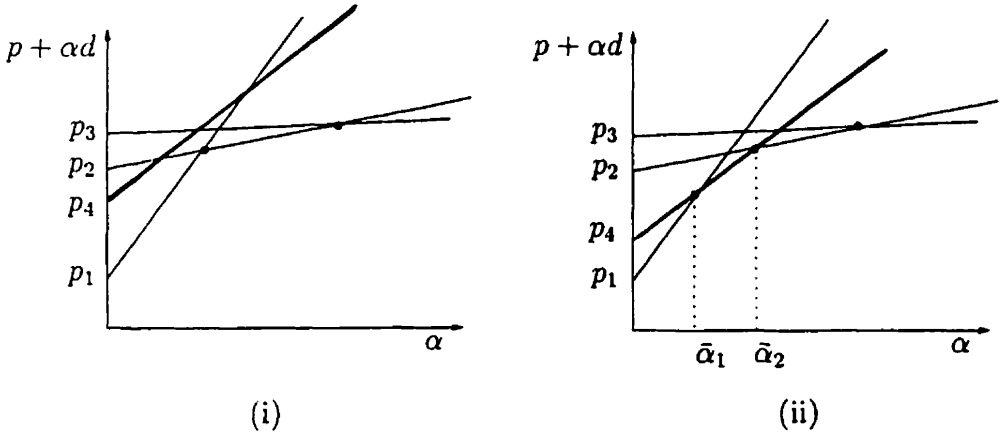


Figure 1.5 – Coût du produit 4

être très élevé. Comme dans l'exemple précédent, une approximation faite à partir de l'évaluation de z en un petit nombre de points bien choisis peut fournir une bonne information, à un coût moindre que pour trouver la solution exacte.

1.3 Formulation mathématique du problème

Dans cette section nous énonçons de façon formelle le problème introduit à la section 1.1. Bien que l'analyse faite au chapitre 2 ne nécessite pas un tel niveau d'abstraction, il est néanmoins utile pour mettre en contexte le problème de décrire celui-ci dans un cadre général. Ceci nous permettra en outre de donner un aperçu des résultats connus dans ce domaine.

1.3.1 Définitions

Plusieurs problèmes mathématiques peuvent être décrits comme étant l'assignation à une fonction f d'une valeur numérique. Par exemple, l'intégration d'une fonction sur un intervalle $[a, b]$ associe à f le scalaire $\int_a^b f(x)\ dx$. Ceci est décrit par l'opérateur $S_1 : X \to \mathbf{R}$, où X est un espace de fonctions auquel appartient f et $S_1(f) = \int_a^b f(x)\ dx$. Plus généralement, on associe à f un élément d'un espace de fonctions. Un exemple de ceci est l'opérateur $S_2 : X \to l^\infty$ qui associe à la fonction f les coefficients de sa série de Taylor. Ici

$$S_2(f) = \left(\frac{f^{(n)}(0)}{n!}\right)_{n \in \mathbf{N}}$$

est un élément de l^∞, l'espace des suites infinies dont la norme uniforme est finie. De façon générale, on décrit un problème mathématique par un opérateur $S : X \to Y$, où X et Y sont des espaces de Banach (c'est-à-dire des espaces vectoriels normés).

Revenons à l'opérateur S_1 et supposons que l'on cherche à approximer la valeur de l'intégrale à partir de l'évaluation de f en n points $x_1, \ldots, x_n$. Cette approximation est l'image d'un opérateur $S^n : X \to \mathbf{R}$ qui, à partir de $f(x_1), \ldots, f(x_n)$, construit une valeur approchée de $S(f) = \int_a^b f(x)\ dx$. L'opérateur S^n est donc une approximation de S en ce sens que $S(f) \approx S^n(f)$ pour $f \in X$. On peut mesurer l'écart entre ces deux valeurs par $|S(f) - S^n(f)|$.

Considérons maintenant le problème qui nous intéresse, soit l'approximation d'une fonction f. Dans ce cas, $X = Y$ et on cherche à approximer l'opérateur identité $S = I : X \to X$. L'approximation se fait dans la norme de X, c'est-à-dire que l'écart entre $I(f) = f$ et $S^n(f)$ est mesuré selon cette norme, par exemple la norme $\mathcal{L}^1$

$$||S^n(f) - f||_1 = \int_a^b |\ S^n(f) - f\ |\ dx,$$

ou la norme uniforme

$$||S^n(f) - f||_\infty = \sup_{x \in [a,b]} | S^n(f)(x) - f(x) |.$$

De façon générale, on peut décrire certains problèmes d'analyse numérique comme étant le calcul de l'approximation d'un opérateur linéaire

$$S : F \subset X \to Y$$

où X et Y sont des espaces de Banach et F est un sous-ensemble, pas nécessairement linéaire, de X.

L'approximation de $S(f)$ se fait au moyen d'une méthode qui repose sur l'information que l'on peut obtenir sur f. Cette information prend la forme d'un autre opérateur $N : X \to \mathbf{R}^n$ avec

$$N(f) = (L_1(f), L_2(f), \ldots, L_n(f))$$

où chaque $L_i : X \to \mathbf{R}$ est une fonctionnelle linéaire. Par exemple, si l'on veut approximer une fonction à partir de sa valeur en n points $x_1, \ldots, x_n$,

$$N(f) = (f(x_1), \ldots, f(x_n)). \tag{1.4}$$

On pourrait aussi utiliser cette même information pour le problème de l'intégration numérique.

La méthode d'approximation, dénotée S^n, est de la forme $S^n = \phi \circ N$ où $\phi : \mathbf{R}^n \to Y$ est une fonction qui, à partir de l'information $N(f)$ donne une approximation de $S(f)$. Ceci est représenté par le diagramme à la figure 1.6. Explicitement, on a

$$S^n(f) = \phi(L_1(f), \ldots, L_n(f)).$$

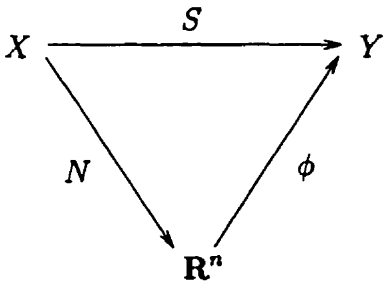


Figure 1.6 – La méthode S^n

Pour un exemple concret, considérons à nouveau le problème d'approximation à l'aide de n points d'évaluation. L'information N est définie comme en 1.4 et la fonction ϕ associe à $(f(x_1), \ldots, f(x_n))$ la fonction linéaire par morceaux dont les points de brisure sont les points $(x_i, f(x_i))$ (voir [16]). Bien sûr, il y aurait beaucoup d'autres possibilités pour ϕ.

Si les fonctionnelles L_k sont fixées d'avance et ne dépendent pas de f, la méthode S^n est dite *non-adaptative* (ou *passive*). Il peut cependant être intéressant de choisir les L_k séquentiellement et d'utiliser l'information déjà calculée donnée par $L_i(f)$ pour $i \leq k-1$ pour choisir la fonctionnelle suivante L_k. Si les L_k dépendent ainsi des valeurs précédentes, S^n est dite *adaptative* (ou *active*). Par exemple, pour le problème d'approximation, on peut avoir $L_k(f) = f(x_k)$, où $x_k = \psi_k(f(x_1), \ldots, f(x_{k-1}))$, pour une suite de fonctions ψ_k. C'est une méthode de ce type, où les points d'évaluation dépendent de l'information obtenue lors des évaluations précédentes, que nous développons au chapitre 2.

Pour mesurer l'efficacité d'une méthode S^n, on définit *l'erreur maximale* de S^n comme étant

$$\Delta_{max}(S^n) = \sup_{f \in F} ||S(f) - S^n(f)||.$$

On interprète celle-ci comme l'erreur dans le «pire cas» pour la méthode S^n. Ceci

n'est pas la seule façon d'évaluer une méthode S^n. On pourrait par exemple mesurer l'erreur moyenne

$$\Delta_{moy}(S^n) = \left(\int_F | S(f) - S^n(f) | \, d\mu \right)^{1/2}$$

où μ est une mesure de probabilité sur l'espace F (voir [24]). Les résultats dans ce contexte peuvent être différents de ceux dans le contexte du pire cas (voir [16]). Dans ce travail, nous ne considérons que ce dernier.

1.3.2 Définition du problème d'approximation

Nous cherchons à approximer une fonction lisse par morceaux, concave, croissante sur [0,1], dont les surgradients en 0 et 1 sont connus. Précisément, nous considérons la classe

$$F_{a,b} \subset C^0[0,1] = \{ f : [0,1] \to [0,1] \mid f \text{ est continue} \},$$

où

$$F_{a,b} = \left\{ f \in C^0[0,1] \;\middle|\; \begin{array}{l} f \text{ concave et croissante,} \\ f(0) = 0, f(1) = 1, \\ f'(0) = a, f'(1) = b \end{array} \right\}$$

Une fonction f concave est nécessairement continue et ses dérivées à droite et à gauche, f'_+ et f'_- existent partout, bien que ça ne soit pas le cas pour la dérivée elle-même. L'exception à ceci est aux extrémités, où $f'_\pm$ peut être non bornée. Nous incluons ce cas en disant que la dérivée est infinie à l'extrémité. Aux points où f' n'est pas définie, on peut calculer un surgradient de f, (voir définition 3 du chapitre 2), aussi dénoté par f'. Notons que ce dernier n'est en général pas unique.

Nous voulons approximer les fonctions de $F_{a,b}$ dans la norme $\mathcal{L}^1$:

$$|| f ||_1 = \int_0^1 | f(x) | \, dx.$$

Ici $S = I$ et $Y = \mathcal{L}^1[0,1]$ est l'espace des fonctions intégrables sur [0,1] dont la norme est finie. L'approximation sera faite à partir de l'information

$$N(f) = (f(x_1), f'(x_1), \ldots, f(x_n), f'(x_n)).$$

Si f n'est pas dérivable en x_k, on supposera que l'on peut calculer un surgradient de f en ce point, que l'on dénotera également par $f'(x_k)$.

Il reste à préciser la fonction ϕ, qui donne l'approximation de f à partir de l'information N. La méthode que nous proposons est une variante de l'algorithme du «sandwich» tel que décrit dans [4]. Celui-ci consiste à construire au moyen des points x_k deux approximations L et U telles que

$$L(x) \leq f(x) \leq U(x) \quad \forall x.$$

Les détails de ces constructions est donné à la section 2.2 du chapitre 2. L'approximation de f est la fonction $U(x)$. On a alors

$$|| U - f ||_1 \leq || U - L ||_1$$

et cette dernière quantité est une borne sur l'erreur maximale.

Le choix des points d'évaluation est fait de façon séquentielle, x_k étant calculé à partir des valeurs de x_{k-1}, $f(x_{k-1})$ et $f'(x_{k-1})$. Dans le but de simplifier le problème de déterminer les meilleurs points x_k, nous avons fixé l'ordre dans lequel ceux-ci sont choisis : on suppose que $x_1 \leq \ldots \leq x_n$ (voir section 2.6 pour une discussion des conséquences de ce choix). Pour obtenir l'expression permettant le calcul des x_k, nous poserons le problème de minimiser l'erreur de l'approximation, étant donné l'ordre

choisi, dans le cadre de la programmation dynamique. Notre approche est inspirée de celle de Bellman et Dreyfus dans [2] pour la recherche du zéro d'une fonction convexe inconnue dont on peut calculer la valeur en tout point d'un intervalle. Du principe d'optimalité qui caractérise la programmation dynamique (voir [2]), nous obtenons une relation de récurrence qui exprime l'erreur maximale, pour l'ordre d'évaluation donné plus haut. Une fois posée cette relation, nous en déterminons une solution analytique qui donne une formule pour le calcul de x_1 (voir théorème 1). De cette formule nous obtenons une méthode S^n_{dyn} pour le calcul successif des x_k et l'approximation des fonctions de $F_{a,b}$. Le théorème 1, principal résultat de ce travail, donne l'erreur maximale de S^n_{dyn} :

$$\Delta_{max}(S^n_{dyn}) = \frac{1}{2(n+1)^2}\left(\frac{(a-1)(1-b)}{a-b}\right). \tag{1.5}$$

Cette borne est optimale pour cette méthode. En effet, on calcule explicitement pour S^n_{dyn} à la section 2.5 l'erreur pour le pire cas. De plus, S^n_{dyn} est optimale quant à l'ordre de l'erreur, $O(n^{-2})$, comme démontré par la fonction $f(x) = \sqrt{1-(x-1)^2}$, dont le graphe est le quart du cercle centré en (1,0). Pour cette fonction on peut montrer que la meilleure approximation linéaire par morceaux est celle dont les points de brisure découpent le quart de cercle en $n+1$ arcs de longueur égale. On observe que dans ce cas l'erreur est d'ordre $O(n^{-2})$. Pour plus de détails à ce sujet, le lecteur pourra consulter [9].

1.3.3 Revue de littérature

Nous avons décrit à la section précédente une procédure adaptative pour l'approximation de fonctions concaves. Une question que l'on peut se poser est la suivante : pour un problème donné, est-il nécessairement avantageux d'utiliser un algorithme adapta-

tif plutôt que passif? À première vue il semble que oui, puisque les premiers utilisent une information plus riche. Cependant, Novak montre dans [16] que la réponse à cette question dépend fortement du problème considéré, en particulier de l'ensemble F. On y prouve que si F est un sous-ensemble convexe et symétrique (c'est-à-dire que $f \in F \Rightarrow -f \in F$) de X alors la réponse est négative lorsque $Y = \mathbf{R}$. Dans ce cas il existe toujours une méthode non-adaptative S_*^n qui est optimale au sens où

$$\Delta_{max}(S_*^n) \leq \Delta_{max}(S^n)$$

quelle que soit S^n, passive ou adaptative. Si Y est quelconque, les méthodes adaptatives diminuent l'erreur maximale d'un facteur d'au plus 2 : il existe toujours une méthode non-adaptative S_*^n telle que

$$\Delta_{max}(S_*^n) \leq 2\Delta_{max}(S^n)$$

quelle que soit S^n (voir [8], [16]). On voit donc que dans ce cas les méthodes passives et adaptatives donnent une erreur du même ordre.

Si F est convexe mais n'est pas symétrique alors la situation est très différente. En effet, dans ce cas il est possible qu'une méthode adaptative soit meilleure que les méthodes passives. On trouve des exemples où c'est le cas dans [17], et dans [18] Novak conjecture que pour l'approximation dans la norme uniforme, les méthodes adaptatives améliorent les méthodes passives par un facteur de n, c'est-à-dire qu'il existe une méthode adaptative S_*^n telle que

$$\Delta_{max}(S_*^n) \leq n\Delta_{max}(S^n)$$

pour toutes les méthodes passives S^n.

Il est à noter que dans le cas qui nous occupe, $F_{a,b}$ est convexe. Cependant $F_{a,b}$ n'est pas symétrique puisque $f \geq 0$ pour tous les f dans cet ensemble.

L'approximation de fonctions a été étudiée dans de nombreux ouvrages pour le cas convexe. Nous nous intéressons plutôt dans ce travail aux fonctions concaves, étant données les applications que nous avons en vue (section 1.2.1), mais les deux problèmes sont équivalents : si f est concave alors $-f$ est convexe. Les résultats du cas convexe s'appliquent donc aussi aux fonctions concaves.

On peut définir plusieurs mesures pour la distance entre les approximations U et L de l'algorithme du sandwich. En plus de la norme $\mathcal{L}^1$, que nous avons choisie pour notre problème, et de la norme uniforme $\mathcal{L}^\infty$ définie à la page 11, il y a la distance de Hausdorff entre deux ensembles :

$$d(U, L) = \max\{\sup_{x \in L} \inf_{y \in U} || y - x ||, \sup_{x \in U} \inf_{y \in L} || y - x ||\}$$

où U et L dénotent ici le graphe des fonctions correspondantes. La première expression entre accolades est la plus grande distance entre un point de L et l'ensemble U. La deuxième expression est interprétée de façon analogue.

L'algorithme du sandwich a été étudié par plusieurs auteurs, qui ont proposé diverses façons de choisir les points servant à construire U et L. Ces méthodes ont toutes en commun, cependant, de choisir à chaque itération un point qui subdivise le sous-intervalle qui présente la plus grande erreur. En ceci, elle diffèrent de notre approche, où l'on ajoute à chaque itération le point qui minimise l'erreur maximale.

Il est à noter que certaines versions de l'algorithme du sandwich proposées dans la littérature ne s'appliquent pas à notre problème. En effet, on suppose que la fonction f à approximer est complètement connue mais qu'il est toutefois préférable de la remplacer par une approximation linéaire par morceaux avec peu de points de brisure. Des exemples sont donnés dans [19]. Dans ce cas le point de subdivision d'un sous-intervalle est la solution d'un problème de minimisation impliquant la fonction f. De telles méthodes ne sont pas utiles, cependant, lorsque la fonction à approximer n'est

pas connue et est coûteuse à évaluer.

Fruhwirth, Burkard et Rote proposent dans [7] trois méthodes de subdivision pour l'approximation avec la distance de Hausdorff et ils donnent une borne pour l'erreur, qui est d'ordre $O(n^{-2})$. Une borne du même ordre est obtenue par Burkard, Hamacher et Rote dans [4], cette fois-ci pour l'approximation dans la norme uniforme. On y propose deux méthodes de subdivision, dont une nécessite la connaissance de la fonction à approximer.

Ces méthodes sont étudiées du point de vue théorique et pratique par Rote dans [19]. On y décrit quatre règles de subdivision, dont deux exigent une connaissance de la fonction. L'approximation est faite en utilisant la norme uniforme, mais l'auteur mentionne que d'autres normes (Hausdorff entre autres), se traitent de la même façon. En plus de donner une nouvelle preuve de la borne d'ordre $O(n^{-2})$ sur l'erreur pour chacune des règles, on compare l'efficacité de celles-ci. On prouve qu'aucune des quatre n'est systématiquement meilleure que les autres, au sens où il existe toujours une «mauvaise» fonction pour laquelle une règle n'est pas efficace mais les autres le sont. Plus généralement ceci est vrai pour toute règle de subdivision. Du point de vue pratique, un fait notable lors des expériences numériques est que l'erreur tend à diminuer brusquement lorsque le nombre de points d'évaluations est une puissance de 2, et plus régulièrement sinon. Ce comportement est semblable à celui de la méthode heuristique INTER définie au chapitre 3.

L'auteur conclut en discutant d'un algorithme du sandwich optimal. Plus précisément, il pose la question suivante :

> Étant donnés une approximation sandwich initiale et un nombre n, quelle est la meilleure stratégie pour choisir les points de subdivision de sorte que la pire erreur après n itérations devienne la plus faible possible ?

Cette question est dans le même esprit que celle que nous avons posée pour notre problème. Comme nous le verrons, nous cherchons avec la méthode S^n_{dyn} non seulement une borne sur l'erreur, mais de plus à minimiser celle-ci dans le pire cas, sachant que l'on a n points à choisir. Au chapitre 3 (section 3.3.2) nous proposons une version de l'algorithme du sandwich où le point de subdivision est obtenu par la formule du théorème 1 et est optimal au sens où l'erreur maximale sur le sous-intervalle est minimisée. Ceci répond à la question posée plus haut, dans le cas où l'erreur est mesurée dans la norme $\mathcal{L}^1$

Les différentes versions de l'algorithme du sandwich proposées dans les ouvrages cités plus haut dépendent toutes de la dérivée (sous-gradient) de la fonction pour la construction de l'approximation L. De même, notre version utilise le surgradient pour la fonction U. Si cette information n'est pas disponible, l'algorithme du sandwich peut être adapté comme le font Yang et Goh dans [25]. Dans cet article, ils trouvent une borne d'ordre $O(n^{-2})$ pour un algorithme du sandwich qui ne dépend pas de la dérivée. Il exige toutefois la connaissance de la fonction à approximer.

Dans [20] Sonnevend étudie dans un contexte plus général l'approximation uniforme de fonctions dont la $(r-1)^{\text{e}}$ dérivée est monotone, pour $r \geq 2$. Il propose trois algorithmes adaptatifs dont l'erreur est d'ordre $O(n^{-r})$, ce qui est optimal. Il prouve également que pour ce problème ces algorithmes sont meilleurs, par un facteur de n, que les algorithmes passifs.

Mentionnons enfin, pour conclure ce bref aperçu, que le même auteur, dans [21] généralise le problème à des fonctions convexes de deux variables et propose des algorithmes adaptatifs qui sont supérieurs aux algorithmes passifs.

CHAPITRE 2

UNE NOUVELLE MÉTHODE ADAPTATIVE

Nous développons dans ce chapitre la méthode S^n_{dyn} introduite au chapitre précédent. À la section 2.1 est décrite en détail la classe $F_{a,b}$. À la section 2.2 sont introduites deux approximations linéaires par morceaux pour les fonctions de $F_{a,b}$, qui serviront à la définition de S^n_{dyn} ainsi qu'à la preuve de la borne sur l'erreur maximale pour cette méthode. On définit à la section 2.3 la relation de récurrence qui servira à construire S^n_{dyn} et on énonce le principal résultat de ce travail, le théorème 1 déjà mentionné, à la section 2.4. On en donne une preuve à la section 2.5. Enfin, les sections 2.6 et 2.7 sont consacrées à une discussion de la portée du théorème et de ses généralisations possibles.

2.1 La classe de fonctions étudiée

Dans cette section nous décrivons de façon précise la classe de fonctions qui sera l'objet de ce chapitre. Rappelons d'abord quelques définitions.

Définition 1. *Une fonction f est*

- croissante *si* $f(s) \leq f(t)$ *lorsque* $s \leq t$.

- concave *si quels que soient* s, t,

$$f(\lambda s + (1-\lambda)t) \geq \lambda f(s) + (1-\lambda)f(t) \quad \forall \lambda \in [0,1].$$

- lisse *si sa dérivée existe en chaque point de son domaine de définition.*
- linéaire par morceaux (LPM) *si elle est définie par*

$$f(t) = m_i t + b_i \quad si \quad t \in [t_{i-1}, t_i]$$

où $\{t_i\}_{i \in I}$ *est une partition de l'intervalle de définition de* f.

Une fonction lisse est toujours continue. Une fonction LPM est continue si pour chaque i, $m_i t_i + b_i = m_{i+1} t_i + b_{i+1}$. Les points t_i sont les *points de brisure* de la fonction LPM f. Par la suite, toutes les fonctions considérées seront continues.

Définition 2. *Une fonction* f *est* normalisée *si* f *est continue, concave, croissante et satisfait* $f(0) = 0$, $f(1) = 1$.

Pour une fonction lisse, $f'(t)$ dénotera la dérivée de f au point t. Dans le cas d'une fonction LPM, $f'(t)$ dénotera un surgradient de f en t. Celui-ci est défini de la façon suivante :

Définition 3. *Le nombre réel* ξ *est un* surgradient *de la fonction concave* f *au point* t *si pour tout* s *dans le domaine de définition de* f

$$f(s) \leq f(t) + \xi(s-t).$$

Cette inégalité est équivalente à

$$\frac{f(s) - f(t)}{s - t} \leq \xi \quad \text{si } s > t$$

$$\frac{f(s) - f(t)}{s - t} \geq \xi \quad \text{si } s < t$$

Ceci est illustré à la figure 2.1.

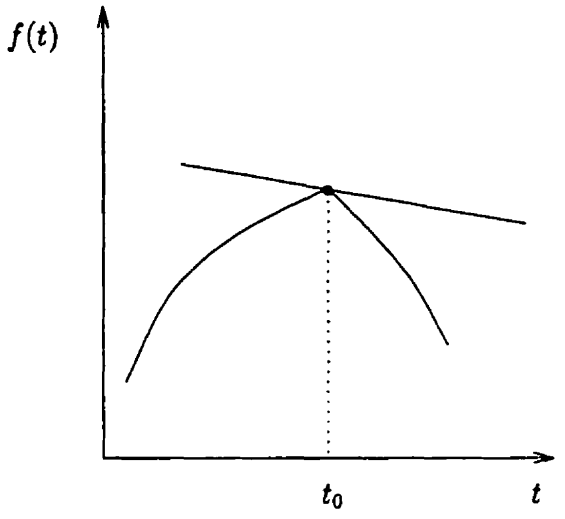


Figure 2.1 – Un surgradient de f en t_0

Notons que si f est lisse alors le surgradient en un point est unique et correspond à la dérivée de f en ce point. À un point de brisure d'une fonction LPM, il y a une infinité de surgradients et l'ensemble de ceux-ci forme un intervalle.

Pour une fonction normalisée f, l'expression *dérivées aux extrémités* désignera les valeurs $f'(0)$ et $f'(1)$. Dans ce qui suit, celles-ci seront habituellement dénotées par a et b respectivement. Enfin, pour simplifier la discussion, nous emploierons le terme «dérivée» pour désigner soit une dérivée, soit un surgradient.

L'étude d'une fonction continue concave croissante générale, par exemple la fonction valeur décrite au chapitre 1, se réduit à l'étude d'une fonction normalisée. En effet, une transformation affine permet de passer de l'une à l'autre. Nous donnons maintenant le détail de cette transformation.

Soit (t_1, u_1) et (t_2, u_2) deux points du plan avec $t_1 < t_2$ et $u_1 < u_2$. On définit les

transformations affines $T_1, T_2 : \mathbf{R} \longrightarrow \mathbf{R}$ par

$$T_1(t) = \frac{t - t_1}{t_2 - t_1} \quad \text{et} \quad T_2(u) = \frac{u - u_1}{u_2 - u_1}.$$

Les inverses de T_1 et T_2 sont

$$T_1^{-1}(t) = (t_2 - t_1)t + t_1 \quad \text{et} \quad T_2^{-1}(u_2 - u_1)u + u_1.$$

On définit aussi une transformation affine T du plan par

$$T(t, u) = (T_1(t), T_2(u))$$

qui satisfait $T(t_1, u_1) = (0, 0)$ et $T(t_2, u_2) = (1, 1)$. L'image par T du rectangle passant par (t_1, u_1) et (t_2, u_2) sur la figure 2.2 est le carré passant par $(0, 0)$ et $(1, 1)$.

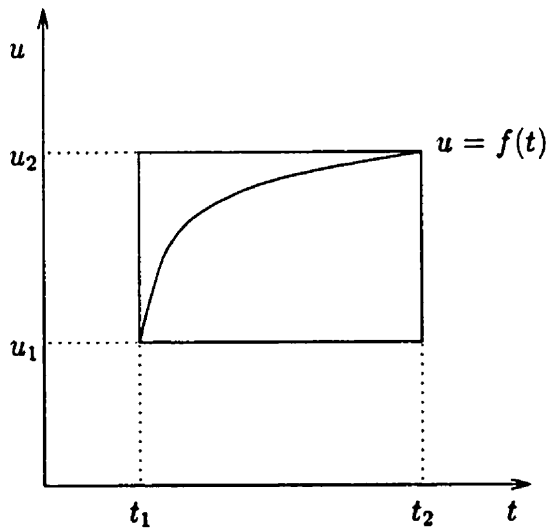


Figure 2.2 – La transformation T

Considérons maintenant une fonction f avec $f(t_1) = u_1$ et $f(t_2) = u_2$. On définit la fonction $\tilde{f}$ par $\tilde{f} = T_2 \circ f \circ T_1^{-1}$, c'est-à-dire

$$\tilde{f}(t) = \frac{f((t_2 - t_1)t + t_1) - u_1}{u_2 - u_1}.$$

La fonction $\tilde{f}$ est concave et croissante, et on a

$$\tilde{f}(0) = \frac{f(t_1) - u_1}{u_2 - u_1} = 0 \quad \text{et} \quad \tilde{f}(1) = \frac{f(t_2) - u_1}{u_2 - u_1} = 1.$$

La transformation T nous fait donc passer d'une fonction concave croissante quelconque à une fonction normalisée. Examinons maintenant l'effet de T sur la dérivée et l'intégrale de f.

Afin de rendre explicite la relation entre la dérivée de $\tilde{f}$ et celle de f, distinguons entre la variable t et son image par T_1 en posant $\tilde{t} = T_1(t)$. Soit ξ un surgradient de f en t. On a

$$\begin{aligned}
\tilde{f}(\tilde{s}) &= \frac{f((t_2 - t_1)\tilde{s} + t_1) - u_1}{u_2 - u_1} \\
&= \frac{f(s) - u_1}{u_2 - u_1} \\
&\leq \frac{f(t) - u_1}{u_2 - u_1} + \xi \frac{(s - t)}{u_2 - u_1} \\
&= \frac{f((t_2 - t_1)\tilde{t} + t_1) - u_1}{u_2 - u_1} + \xi \frac{(t_2 - t_1)\tilde{s} + t_1 - (t_2 - t_1)\tilde{t} - t_1}{u_2 - u_1} \\
&= \tilde{f}(\tilde{t}) + \xi \left(\frac{t_2 - t_1}{u_2 - u_1} \right) (\tilde{s} - \tilde{t}).
\end{aligned}$$

Ceci montre que $\xi \left(\frac{t_2 - t_1}{u_2 - u_1} \right)$ est un surgradient de $\tilde{f}$. La relation entre la dérivée (surgradient) de f en t et celle de $\tilde{f}$ en $\tilde{t}$ est donc donnée par

$$\tilde{f}'(\tilde{t}) = \frac{t_2 - t_1}{u_2 - u_1} f'(t).$$

En particulier, si $f'(t_1) = \mu_1$ et $f'(t_2) = \mu_2$ alors les dérivées de $\tilde{f}$ aux extrémités sont

$$a = \left(\frac{t_2 - t_1}{u_2 - u_1} \right) \mu_1 \quad \text{et} \quad b = \left(\frac{t_2 - t_1}{u_2 - u_1} \right) \mu_2.$$

Ceci nous sera utile à la section 2.3 pour la formulation précise de notre problème.

Examinons finalement la relation entre l'intégrale de f et celle de $\tilde{f}$. En utilisant la notation introduite plus haut,

$$\begin{aligned}\int_{\tilde{c}}^{\tilde{d}} \tilde{f}(\tilde{t})\, d\tilde{t} &= \int_{\tilde{c}}^{\tilde{d}} \frac{f((t_2-t_1)\tilde{t}+t_1)-u_1}{u_2-u_1}\, d\tilde{t} \\ &= \frac{1}{(t_2-t_1)(u_1-u_2)} \int_c^d (f(t)-u_1)\, dt.\end{aligned}$$

Si R est une région bornée par les graphes de f et g entre c et d et $\tilde{R} = T(R)$ on a

$$\begin{aligned}\text{aire}(\tilde{R}) &= \int_{\tilde{c}}^{\tilde{d}} (\tilde{f}(\tilde{t}) - \tilde{g}(\tilde{t}))\, d\tilde{t} \\ &= \frac{1}{(t_2-t_1)(u_2-u_1)} \int_c^d (f(t)-g(t))\, dt \\ &= \frac{1}{(t_2-t_1)(u_2-u_1)} \text{aire}(R)\end{aligned}$$

ou encore

$$\text{aire}(R) = (t_2-t_1)(u_2-u_1) \times \text{aire}(\tilde{R}). \tag{2.1}$$

Ce résultat sera utilisé à la section 2.3.

2.2 Les approximations L et U

Nous définissons dans cette section les approximations inférieures et supérieures, L et U, de notre algorithme du sandwich pour une fonction normalisée f. Supposons que l'on ait calculé la valeur de f et f' en n points $t_1, \ldots, t_n$ de l'intervalle [0,1]. L'approximation L de f est la spline linéaire passant par les points $(t_i, f(t_i))$, c'est-à-dire

$$L(x) = \frac{f(t_{i-1})(t-t_i)}{t_{i-1}-t_i} + \frac{f(t_i)(t-t_{i-1})}{t_i-t_{i-1}} \quad \text{si} \quad t \in [t_{i-1}, t_i].$$

La fonction L illustrée sur la figure 2.2.

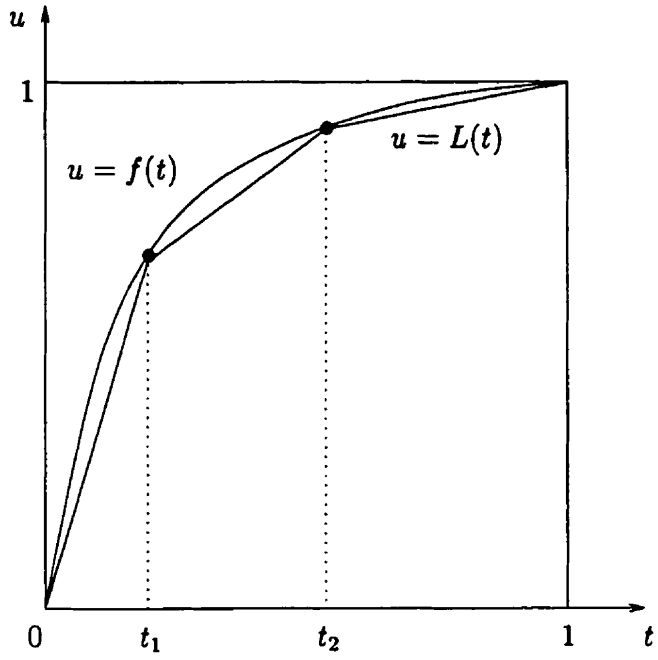


Figure 2.3 – Approximation L avec $n = 2$.

On remarque que puisque f est concave, on a

$$L(t) \leq f(t).$$

L'approximation U est construite comme suit. En chacun des points t_i, $f'(t_i)$ est la pente d'une droite tangente au graphe de f en t_i. Cette droite a pour équation

$$u = f'(t_i)(t - t_i) + f(t_i).$$

L'abscisse du point d'intersection des tangentes en deux points consécutifs t_{i-1} et t_i est

$$\bar{t}_i = \frac{f(t_{i-1}) - f(t_i) + f'(t_i)t_i - f'(t_{i-1})t_{i-1}}{f'(t_i) - f'(t_{i-1})}.$$

On définit U par

$$U(t) = f'(t_i)(t - t_i) + f(t_i) \quad \text{si} \quad t \in [\bar{t}_{i-1}, \bar{t}_i].$$

La fonction S est illustrée à la figure 2.4. Notons que s'il y a n points d'évaluation, on obtient ainsi $n + 1$ points $\bar{t}_i$.

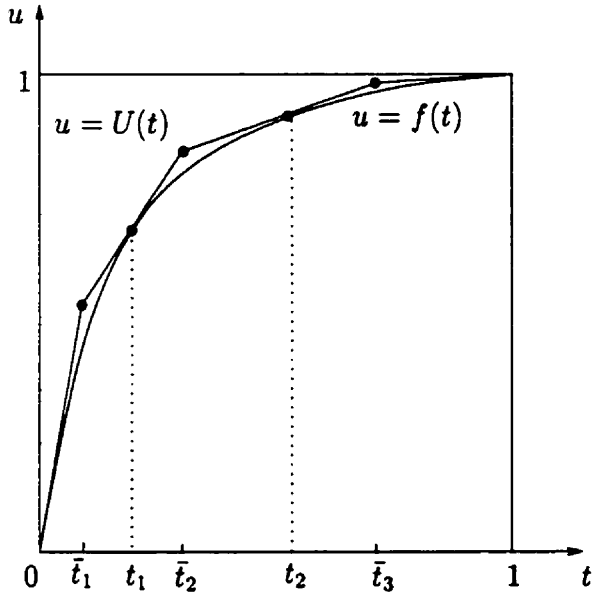


Figure 2.4 – Approximation U avec $n = 2$.

Puisque f est concave, on a cette fois $f(t) \leq U(t)$. Comme nous l'avons noté précédemment,

$$\int_0^1 (U(t) - f(t))\, dt \leq \int_0^1 (U(t) - L(t))\, dt$$

c'est-à-dire que l'erreur commise en approximant l'intégrale de f par celle de U est bornée supérieurement par $\int_0^1 (U - L)$. Cette dernière quantité est l'aire de la région hachurée sur la figure 2.5. Par la suite, c'est cette intégrale que nous chercherons à minimiser en choisissant les points t_i de façon appropriée. Notons que L et U dépendent du nombre de points n, mais pour alléger la notation, cette dépendence ne sera pas indiquée explicitement.

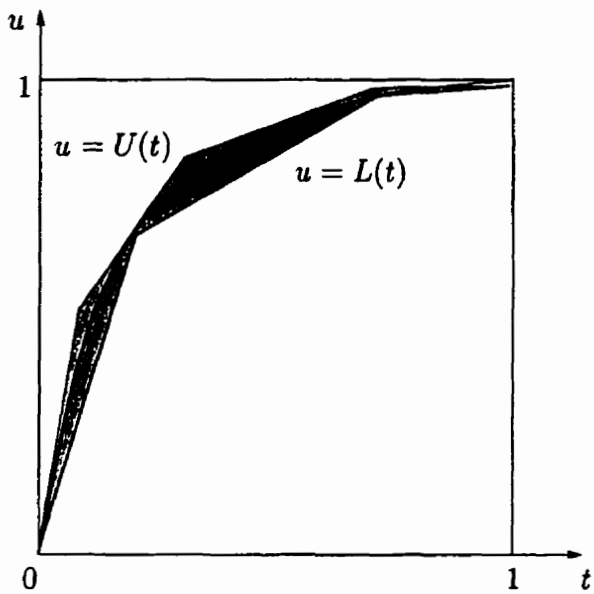


Figure 2.5 - $\int_0^1 (U - L)$.

2.3 Définition de l'erreur $\mathcal{E}_n$

2.3.1 Stratégie d'évaluation

Nous définissons dans cette section la fonction $\mathcal{E}_n$ dont l'évaluation fera l'objet de la section 2.4. Nous verrons que $\mathcal{E}_n$ donne une borne supérieure pour l'intégrale $\int_0^1 (U - L)$ décrite à la section précédente. Donnons un nom à cette intégrale :

Définition 4. *Étant donnée une fonction f et des points d'évaluation $t_1, \ldots, t_n$,* l'erreur maximale *commise en approximant l'intégrale de f par celle de U est la quantité* $\int_0^1 (U(t) - L(t))\, dt$.

Nous cherchons à minimiser l'erreur maximale en choisissant judicieusement les

points où la fonction f sera évaluée. Précisons qu'il ne s'agit pas ici de déterminer a priori tous les points t_i. Nous voulons plutôt, dans l'esprit de la programmation dynamique, utiliser l'information recueillie au fur et à mesure des évaluations pour choisir l'emplacement des points suivants. Il faut garder à l'esprit la motivation de ce travail, dont nous avons discuté au chapitre 1 : f est une fonction dont on peut évaluer la valeur et la dérivée en tout point, mais sur laquelle nous ne disposons pas d'autre information.

Le choix des points d'évaluation sera donc un processus séquentiel, chaque point apportant de l'information permettant de choisir le suivant. Pour réaliser ce processus il faut une stratégie d'évaluation, c'est-à-dire le choix d'un ordre dans lequel les points seront déterminés. Nous avons décidé de la stratégie suivante : à chaque fois qu'un point est déterminé, le point suivant sera choisi *à droite* de celui-ci. C'est donc dire que l'on choisit n points de gauche à droite sur l'intervalle [0,1]. Il est important de noter que cette stratégie n'est pas nécessairement optimale et que d'autres choix sont possibles. Notre choix est motivé par le fait qu'il est relativement simple d'exprimer mathématiquement le problème de minimiser l'erreur maximale avec cette stratégie. Plus important encore, il s'avère également que l'on peut trouver une solution analytique simple à ce problème, ce qui facilite les applications pratiques. Nous avons inclus à la section 2.6 une discussion portant sur la stratégie d'évaluation optimale. Nous nous concentrerons dans cette section sur la stratégie *gauche à droite*.

2.3.2 Cas $n = 0$

Soit f une fonction normalisée. On connaît la valeur de f aux extrémités de l'intervalle [0,1] : $f(0) = 0$ et $f(1) = 1$. Les dérivées (surgradients) aux extrémités

sont également connues : selon la convention adoptée plus haut

$$f'(0) = a \quad \text{et} \quad f'(1) = b.$$

Puisque f est concave et croissante, on a

$$1 \leq a < \infty \quad \text{et} \quad 0 \leq b \leq 1.$$

Le cas extrême $a = b = 1$ correspond à la fonction $f(t) = t$ et est très simple à analyser. Dans ce cas $U = L$ et $\int_0^1 (U - L) = 0$ et sans même évaluer la fonction on connaît son intégrale. Nous supposerons donc à partir de maintenant que $a > 1$ et $b < 1$. Si $n = 0$, il n'y a aucune évaluation à faire et la seule information dont on dispose est la valeur de f et f' aux extrémités. Dans ce cas

$$L(t) = t$$

et

$$U(t) = \begin{cases} at & \text{si} \quad t \in [0, \frac{1-b}{a-b}] \\ \\ bt + 1 - b & \text{si} \quad t \in [\frac{1-b}{a-b}, 1] \end{cases}.$$

Ceci est illustré à la figure 2.6.

Géométriquement, l'erreur maximale est l'aire du triangle OPK. Dans ce cas particulier, $\int_0^1 (U - L)$ ne dépend pas de points d'évaluation et minimiser l'erreur maximale revient à calculer cette aire. Nous dénoterons sa valeur par $\mathcal{E}_0(a, b)$. On a donc

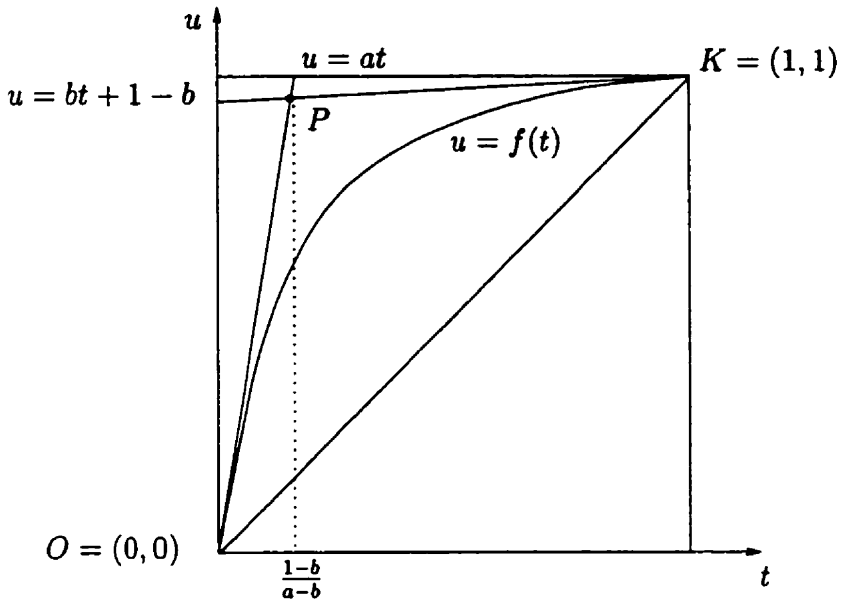


Figure 2.6 – Cas $n = 0$.

$$\begin{aligned}
\mathcal{E}_0(a,b) &= \frac{1}{2}\left|\det\begin{pmatrix} \frac{1-b}{a-b} & a\frac{1-b}{a-b} \\ 1 & 1 \end{pmatrix}\right| \\
&= \frac{1}{2}\frac{1-b}{a-b}\begin{vmatrix} 1 & a \\ 1 & 1 \end{vmatrix} \\
&= \frac{1}{2}\frac{(1-b)(a-1)}{a-b}
\end{aligned}$$

2.3.3 Formulation pour le cas $n = 1$

Examinons maintenant le cas où il y a une seule évaluation à effectuer. La situation est résumée à la figure 2.7. Dans ce qui suit (x, v) dénotera un point fixé et les axes de coordonnés seront identifiés comme auparavant par les variables t et u.

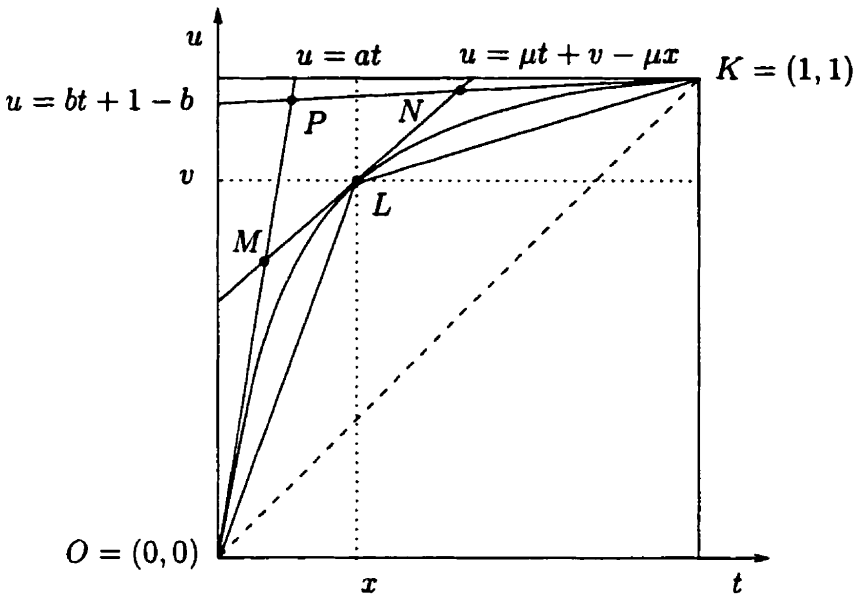


Figure 2.7 - Cas $n = 1$.

En évaluant f en x on trouve une valeur $v = f(x)$. Puisque le graphe de f est contenu dans le triangle OPK, on a

$$x \leq v \leq ax \qquad \text{si} \quad x \in [0, \tfrac{1-b}{a-b}]$$

$$x \leq v \leq bx + 1 - b \quad \text{si} \quad x \in [\tfrac{1-b}{a-b}, 1]$$

On trouve également une valeur $\mu = f'(x)$ qui est un surgradient de f en (x, v). Géométriquement, μ doit être inférieure à la pente du segment OL et supérieure à celle du segment LK. Autrement dit

$$\frac{1-v}{1-x} \leq \mu \leq \frac{v}{x}.$$

Nous arrivons ici à un point crucial de la formulation du problème. Nous cherchons à minimiser la somme des aires des triangles OML et LNK de la figure 2.7. On re-

marque que, une fois déterminés x et v, l'aire de OML correspond à l'erreur maximale sur $[0, x]$ étant données les dérivées aux extrémités a et μ, plus celle sur $[x, 1]$ étant données μ et b. En utilisant la transformation T de la section 2.1 on peut exprimer ceci en terme de la fonction $\mathcal{E}_0$ définie plus haut : les dérivées a et μ deviennent

$$\frac{v}{x}a \text{ et } \frac{v}{x}\mu$$

et $\mathcal{E}_0\left(\frac{v}{x}a, \frac{v}{x}\mu\right)$ est l'aire de l'image par T de OML. Pour avoir l'aire de OML lui-même, il faut utiliser la relation 2.1 de la section 2.1, ce qui donne

$$\text{aire}(OML) = xv\mathcal{E}_0\left(\frac{v}{x}a, \frac{v}{x}\mu\right).$$

De façon semblable, on trouve

$$\text{aire}(LNK) = (1-x)(1-v)\mathcal{E}_0\left(\frac{1-v}{1-x}\mu, \frac{1-v}{1-x}b\right).$$

Ne connaissant pas d'avance v et μ, on doit envisager le pire cas possible, c'est-à-dire les valeurs de v et μ qui maximisent la somme des aires des deux triangles, étant données les contraintes. Ce pire cas s'exprime, pour x fixé, par

$$\max_v \max_\mu \left\{ xv\mathcal{E}_0\left(\frac{v}{x}a, \frac{v}{x}\mu\right) + (1-x)(1-v)\mathcal{E}_0\left(\frac{1-v}{1-x}\mu, \frac{1-v}{1-x}b\right)\right\}.$$

Enfin, on veut minimiser cette quantité en choisissant x de la meilleure façon possible. Si on dénote le minimum par $\mathcal{E}_1(a, b)$ on peut écrire

$$\mathcal{E}_1(a,b) = \min_x \max_v \max_\mu \left\{ xv\mathcal{E}_0\left(\tfrac{v}{x}a, \tfrac{v}{x}\mu\right) + (1-x)(1-v)\mathcal{E}_0\left(\tfrac{1-v}{1-x}\mu, \tfrac{1-v}{1-x}b\right)\right\}.$$

Nous avons donc trouvé une formule récursive pour calculer la plus petite borne supérieure sur l'erreur maximale lorsqu'il y a une seule évaluation à faire. Le raisonnement permettant de trouver cette formule se généralise maintenant à une valeur quelconque de n.

2.3.4 Cas général

Lorsqu'il y a $n > 1$ évaluations à faire, la situation géométrique est très semblable au cas $n = 1$. Nous cherchons toujours à minimiser la somme des erreurs maximales à droite et à gauche du premier point x, mais cette fois-ci il reste encore $n - 1$ points à placer à droite de x. Avant de donner une formule générale, nous devons d'abord définir $\mathcal{E}_n$.

Définition 5. *Étant donnée une fonction telle que $f'(0) = a$ et $f'(1) = b$, $\mathcal{E}_n(a, b)$ est la valeur minimum de l'erreur maximale, sachant que l'on peut effectuer n évaluations de f selon la stratégie gauche à droite.*

On peut maintenant écrire une formule récursive pour $\mathcal{E}_n$. À gauche du premier point x, il n'y a plus d'évaluations à faire et l'erreur maximale est donnée par $\mathcal{E}_0$. À droite de x, il reste $n - 1$ évaluations et l'erreur maximale est exprimée par $\mathcal{E}_{n-1}$. Explicitement,

$$\mathcal{E}_n(a, b) = \min_x \max_v \max_\mu \left\{ xv\mathcal{E}_0\left(\tfrac{v}{x}a, \tfrac{v}{x}\mu\right) + (1-x)(1-v)\mathcal{E}_{n-1}\left(\tfrac{1-v}{1-x}\mu, \tfrac{1-v}{1-x}b\right)\right\}. \quad (2.2)$$

sous les contraintes

$$0 \le x \le 1$$

$$x \le v \le ax \quad \text{si} \quad 0 \le x \le \frac{1-b}{a-b}$$

$$x \le v \le bx + 1 - b \quad \text{si} \quad \frac{1-b}{a-b} \le x \le 1$$

$$\frac{1-v}{1-x} \le \mu \le \frac{v}{x}.$$

Nous nous proposons dans les sections qui suivent de résoudre explicitement ce problème.

2.4 Une formule pour les points optimaux

Nous énonçons dans cette section le principal résultat du chapitre 2. Il s'agit d'une formule pour la valeur de $\mathcal{E}_n(a,b)$ ainsi que pour la solution x^* du problème de minimisation qui définit $\mathcal{E}_n$. Cette solution x^* est le premier point du processus séquentiel d'évaluation de f. Il est optimal, rappelons-le, au sens où il minimise la borne supérieure sur l'erreur maximale pour la stratégie gauche à droite.

Théorème 1. *L'erreur minimale est égale à*

$$\mathcal{E}_n(a,b) = \frac{1}{2(n+1)^2}\frac{(a-1)(1-b)}{(a-b)}$$

et est atteinte au point

$$x^* = \frac{1}{(n+1)^2}\left(1 + 2n\frac{1-b}{a-b}\right).$$

Notons que la première partie du théorème peut s'écrire

$$\mathcal{E}_n(a,b) = \frac{1}{(n+1)^2}\mathcal{E}_0(a,b)$$

La preuve de ce résultat est donnée à la section suivante. Comme elle est passablement longue, nous donnons d'abord ici un résumé des ses principaux éléments.

Résumé de la preuve

- La preuve se fait par récurrence sur n. Le cas $n = 0$ déjà analysé en constitue la base.
- Dans l'expression de $\mathcal{E}_n(a,b)$ on substitue, par l'hypothèse de récurence, la valeur de $\mathcal{E}_{n-1}$ donnée par le théorème. En considérant d'abord x et $v = f(x)$ fixés, on obtient ainsi une fonction ϕ de la variable μ. Cette fonction est illustrée à la figure 2.8.

- Sur l'intervalle $[\frac{1-v}{1-x}, \frac{v}{x}]$ qui nous intéresse, ϕ atteint son maximum soit en un point critique, soit à l'une des extrémités de l'intervalle. Ces trois possibilités donnent lieu à trois cas distincts, qui dépendent de la région du carré unitaire à laquelle appartient (x, v). Ces régions sont données à la figure 2.9. Le maximum μ_{max} est atteint au point critique si $(x, v) \in$ I, à l'extrémité droite si $(x, v) \in$ II et à l'extrémité gauche si $(x, v) \in$ III.
- En évaluant ϕ en μ_{max} on obtient, x étant toujours fixé, une fonction $\psi = \phi(\mu_{max})$ de la variable v. Pour chaque cas, on trouve le maximum v_{max} de ψ.
- Enfin, en évaluant ψ en v_{max} on trouve une fonction $\mathcal{R}_n = \psi(v_{max})$ de x. Il ne reste plus qu'a minimiser θ sur l'intervalle [0,1].
- Chacun des cas I, II, III se divise en plusieurs sous-cas, qu'il faut examiner en détail. On constate que si $(x, v) \in$ I alors on trouve les valeurs annoncées par le théorème pour $\mathcal{E}_n(a, b)$ et x^*. Le reste de la preuve consiste alors à montrer qu'on ne peut améliorer ce résulat lorsque (x, v) appartient à II ou III.

2.5 Preuve du théorème

La preuve de la partie (1) du théorème se fait par récurrence sur n. La partie (2) découle de la preuve de (1). Pour $n = 0$, on a montré en 2.3.2 que

$$\mathcal{E}_0(a, b) = \frac{(a-1)(1-b)}{2(a-b)}$$

et donc le résultat est vérifié. Supposons que

$$\mathcal{E}_{n-1}(a, b) = \frac{(a-1)(1-b)}{2n^2(a-b)}$$

et montrons que

$$\mathcal{E}_n(a, b) = \frac{(a-1)(1-b)}{2(n+1)^2(a-b)}.$$

Pour simplifier la discussion posons, pour a et b fixés,

$$\mathcal{R}_n(x) = \max_v \max_\mu \left\{ xv\mathcal{E}_0\left(\tfrac{x}{v}a, \tfrac{x}{v}\mu\right) + (1-x)(1-v)\mathcal{E}_{n-1}\left(\tfrac{1-x}{1-v}\mu, \tfrac{1-x}{1-v}b\right)\right\}. \qquad (2.3)$$

En prenant le minimum sur x de la fonction $\mathcal{R}_n$, on obtient la valeur de $\mathcal{E}_n(a,b)$:

$$\mathcal{E}_n(a,b) = \min_x \mathcal{R}_n(x). \qquad (2.4)$$

Maintenant fixons x et v et posons

$$\phi(\mu) = 2\left[xv\mathcal{E}_0\left(\frac{x}{v}a, \frac{x}{v}\mu\right) + (1-x)(1-v)\mathcal{E}_{n-1}\left(\frac{1-x}{1-v}\mu, \frac{1-x}{1-v}b\right)\right].$$

Le facteur 2 apparaît pour permettre une simplification ultérieure. Développons cette expression. On a

$$\begin{aligned} xv\mathcal{E}_0\left(\frac{x}{v}a, \frac{x}{v}\mu\right) &= xv\left[\frac{(\frac{x}{v}a-1)(1-\frac{x}{v}\mu)}{(\frac{x}{v}a-\frac{x}{v}\mu)}\right] \\ &= xv\left[\frac{\frac{1}{v^2}(ax-v)(v-\mu x)}{\frac{x}{v}(a-b)}\right] \\ &= \frac{(ax-v)(v-\mu x)}{a-\mu} \end{aligned}$$

En utilisant l'hypothèse de récurrence

$$\begin{aligned} &(1-x)(1-v)\mathcal{E}_{n-1}\left(\frac{1-x}{1-v}\mu, \frac{1-x}{1-v}b\right) \\ &= (1-x)(1-v)\frac{\left(\frac{1-x}{1-v}\mu - 1\right)\left(1-\frac{1-x}{1-v}b\right)}{2n^2\left(\frac{1-x}{1-v}\mu - \frac{1-x}{1-v}b\right)} \\ &= (1-x)(1-v)\frac{(\frac{1}{1-v})^2[\mu-(1-v)][(1-v)-(1-x)b]}{2n^2(\frac{1-v}{1-x})(1-x)(\mu-b)} \\ &= \frac{[(1-x)\mu-(1-v)][(1-v)+(1-x)b]}{2n^2(\mu-b)}. \end{aligned}$$

On a donc

$$\phi(\mu) = \frac{(ax - v)(v - \mu x)}{a - \mu} + \frac{[(1-x)\mu - 1 + v][bx + 1 - b - v]}{n^2(\mu - b)}.$$

Pour simplifier l'écriture, posons

$$A = ax - v \quad \text{et} \quad B = 1 - v - (1-x)b.$$

Notons que $A, B \geq 0$ et que A et B ne dépendent pas de μ. On obtient

$$\phi(\mu) = A\frac{v - \mu x}{a - \mu} + B\frac{(1-x)\mu - (1-v)}{n^2(\mu - b)}.$$

Enfin, pour bien mettre en évidence les caractéristiques de la fonction ϕ, on écrit

$$\phi(\mu) = xA\frac{\frac{v}{x} - \mu}{a - \mu} + (1-x)\frac{B}{n^2}\frac{\mu - \frac{1-v}{1-x}}{\mu - b}.$$

Analysons maintenant cette fonction. On supposera que $A, B > 0$. Les cas particuliers où ceci n'est pas vérifié seront traités à la fin. Rappelons que

$$b \leq \frac{1-v}{1-x} \leq \mu \leq \frac{v}{x} \leq a.$$

Nous nous intéressons à ce qui se passe sur l'intervalle $[b, a]$. La dérivée de ϕ (par rapport à μ) est donnée par

$$\phi'(\mu) = -\frac{A^2}{(a-\mu)^2} + \frac{1}{n^2}\frac{B^2}{(\mu - b)^2}$$

et on a $\phi'(\mu) = 0 \Leftrightarrow \dfrac{A}{a-\mu} = \pm\dfrac{B}{n(\mu - b)}$. Les deux solutions de cette équation sont

$$\mu^+ = \frac{nAb + Ba}{nA + B} \quad \text{et} \quad \mu^- = \frac{nAb - Ba}{nA - B}.$$

En substituant ces valeurs dans l'expression de la dérivée seconde

$$\phi''(\mu) = \frac{-2A^2}{(a-\mu)^3} - \frac{2B^2}{n^2(\mu - b)^3}$$

on trouve

$$\phi''(\mu^+) = \frac{-2(nA+B)^4}{n^3(a-b)^3AB} \leq 0 \quad \text{et} \quad \phi''(\mu^-) = -\phi''(\mu^+) \geq 0.$$

Le point μ^+ correspond donc à un maximum local et μ^- à un minimum. Ce minimum n'est pas dans l'intervalle (b,a) car si $nA - B \geq 0$ alors

$$\mu^- = \frac{nAb - Ba}{nA - B} \leq b \Leftrightarrow -Ba \leq -Bb \Leftrightarrow b \leq a$$

et si $nA - B \leq 0$

$$\mu^- = \frac{nAb - Ba}{nA - B} \geq a \Leftrightarrow nAb \leq nA \Leftrightarrow b \leq a.$$

Dans les deux cas la dernière inégalité est toujours vérifiée. Le maximum μ^+, lui, doit appartenir à (b,a), étant combinaison convexe de a et b. Le graphe de ϕ est donné à la figure 2.8. Celle-ci illustre le cas particulier où $\mu^- < b$ et $\mu^+ \in [\frac{1-v}{1-x}, \frac{v}{x}]$. Les autres cas, qui diffèrent par la position de μ^+ et μ^-, sont semblables. Les cas où $nA - B = 0$ ou $nA + B = 0$ seront traités à la fin.

Nous cherchons le maximum de ϕ sur $\left[\frac{1-v}{1-x}, \frac{v}{x}\right]$. Trois situations peuvent se produire : μ^+ appartient à l'intervalle et c'est le maximum, ou le maximum est atteint à l'une des deux extrémités. Déterminons sous quelles conditions on a $\frac{1-v}{1-x} \leq \mu^+ \leq \frac{v}{x}$. D'abord,

$$\mu^+ \leq \frac{v}{x}$$

$$\Leftrightarrow \quad \frac{nAb + Ba}{nA + B} \leq \frac{v}{x}$$

$$\Leftrightarrow \quad nA(bx - v) + B(ax - v) \leq 0$$
$$\Leftrightarrow \quad nA(bx - v) + (1 - v + bx - b)A \leq 0$$
$$\Leftrightarrow \quad A((n+1)bx - (n+1)v + 1 - b) \leq 0$$

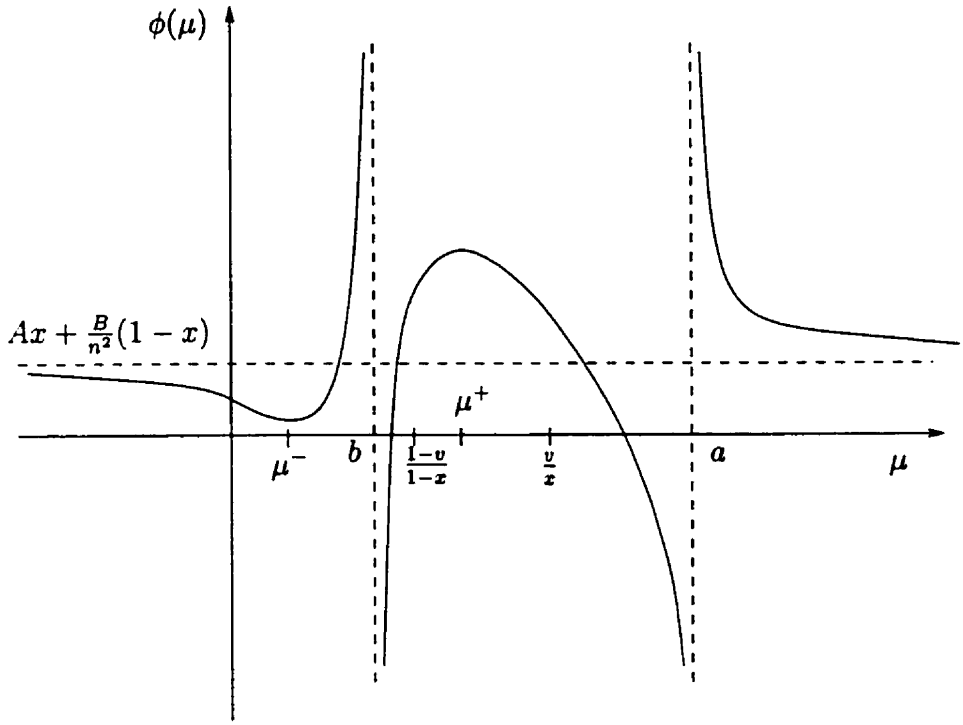


Figure 2.8 - La fonction ϕ.

Puisque $A \geq 0$, on a donc

$$\mu^+ \leq \frac{v}{x} \Leftrightarrow v \geq bx + \frac{1-b}{n+1}.$$

De façon semblable,

$$\mu^+ \geq \frac{1-v}{1-x}$$

$$\begin{aligned}
&\Leftrightarrow \quad \frac{nAb + Ba}{nA + B} \geq \frac{1-v}{1-x} \\
&\Leftrightarrow \quad B(a - ax - 1 + v) - nA(1 - v + bx - b) \geq 0 \\
&\Leftrightarrow \quad B(a - ax - 1 + v) - n(ax - v)B \geq 0 \\
&\Leftrightarrow \quad B(a - 1 - (n+1)ax + (n+1)v) \geq 0
\end{aligned}$$

et, comme $B \geq 0$,

$$\mu^+ \geq \frac{1-x}{1-v} \Leftrightarrow v \geq ax + \frac{1-a}{n+1}.$$

L'information recueillie jusqu'ici se trouve résumée sur la figure 2.9.

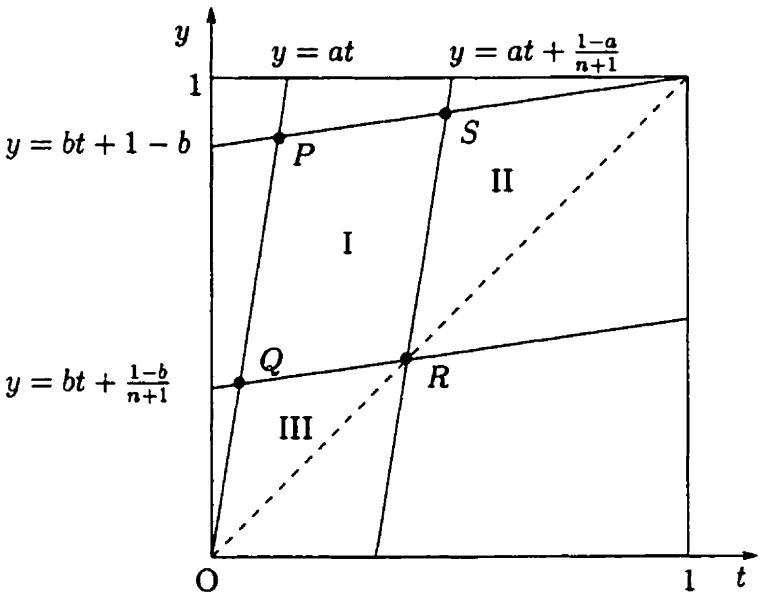


Figure 2.9 – Trois cas pour μ.

Notons les coordonnées des points qui y sont indiqués :

$$P = \left(\frac{1-b}{a-b}, a\frac{1-b}{a-b}\right)$$

$$Q = \left(\frac{1-b}{(n+1)(a-b)}, \frac{a(1-b)}{(n+1)(a-b)}\right)$$

$$R = \left(\frac{1}{n+1}, \frac{1}{n+1}\right)$$

$$S = \left(\frac{a-(n+1)b+n}{(n+1)(a-b)}, \frac{na(1-b)}{(n+1)(a-b)} + \frac{1}{n+1}\right).$$

Si (x, v) appartient à la région I alors $v \geq bx + \frac{1-b}{n+1}$ et $v \geq ax + \frac{1-a}{n+1}$, donc $\mu^+ \in [\frac{1-v}{1-x}, \frac{v}{x}]$ et le maximum de ϕ est atteint en $\mu_{max} = \mu^+$. Si (x, v) appartient à la région II alors $v \leq ax + \frac{1-a}{n+1}$ donc $\mu^+ \leq \frac{1-v}{1-x}$ et ϕ est décroissante sur $[\frac{1-v}{1-x}, \frac{v}{x}]$. La fonction ϕ atteint donc son maximum en $\mu_{max} = \frac{1-v}{1-x}$. Enfin, si (x, v) appartient à la région III alors $v \leq bx + \frac{1-b}{n+1}$ donc $\mu^+ \geq \frac{v}{x}$, ϕ est croissante et le maximum est en $\mu_{max} = \frac{v}{x}$. Nous examinons maintenant chacun de ces trois cas.

Cas $(x, v) \in \mathbf{I}$

Dénotons par P_x, Q_x, R_x et S_x les abscisses des points donnés plus haut. On a $Q_x \leq x \leq S_x$. Posons aussi

$$\begin{aligned} v_0 &= \max\{bx + \tfrac{1-b}{n+1}, ax + \tfrac{1-a}{n+1}\} \\ v_1 &= \min\{ax, bx + 1 - b\}. \end{aligned}$$

On a alors $v_0 \leq v \leq v_1$. Soit $\psi(v) = n^2(a-b)\phi(\mu_{max}) = n^2(a-b)\phi(\mu^+)$. Pour obtenir la valeur de $\psi(v)$, calculons

$$\begin{aligned} xA\frac{\frac{v}{x} - \mu^+}{a - \mu^+} &= xA\frac{\frac{v}{x} - \frac{nAb+Ba}{nA+B}}{a - \frac{nAb+Ba}{nA+B}} \\ &= \frac{A[v(nA+B) - x(nAb+Ba)]}{nAa - nAb} \\ &= \frac{n[v(nA+B) - x(nAb+Ba)]}{n^2(a-b)} \end{aligned} \tag{2.5}$$

$$
\begin{aligned}
(1-x)\frac{B}{n^2}\frac{\mu^+ - \frac{1-v}{1-x}}{b - \frac{1-v}{1-x}} &= (1-x)\frac{B}{n^2}\frac{\frac{nAb+Ba}{nA+B} - \frac{1-v}{1-x}}{\frac{nAb+Ba}{nA+B} - b} \\
&= \frac{B}{n^2}\frac{[(nAb+Ba)(1-x) - (nA+B)(1-v)]}{Ba - Bb} \\
&= \frac{(nAb+Ba)(1-x) - (nA+B)(1-v)}{n^2(a-b)} \qquad (2.6)
\end{aligned}
$$

Après simplification on obtient

$$
\begin{aligned}
\psi(v) &= n^2(a-b)[\ (2.5) + (2.6)\ ] \\
&= A(v - bx) - (2n+1)AB + B(a-1) \\
&= -(n+1)^2v^2 + [(a+b)(n+1)^2x + 2(n+1) - (2n+1)b - a]v \\
&\quad +(bx - b + 1)(a - 1 - (2n+1)ax) - n^2abx^2
\end{aligned}
$$

La fonction ψ est quadratique (en v) et son coefficient dominant $-(n+1)^2$ est négatif. En dérivant par rapport à v, on trouve

$$\psi'(v) = -2(n+1)^2v + (n+1)^2(a+b)x + 2(n+1) - a - (2n+1)b$$

et l'équation $\psi'(v) = 0$ a pour solution

$$\bar{v} = \frac{a+b}{2}x + \frac{2(n+1) - a - (2n+1)b}{2(n+1)^2}.$$

Cette valeur correspond au maximum de ψ. Soit $\mathcal{D}$ la droite définie par cette équation. Il faut maintenant déterminer si $\bar{v}$ est dans l'intervalle que nous considérons, c'est-à-dire si $(x, \bar{v}) \in$ I. Ceci dépend de x, a, b et n, et il y a deux situations possibles, tel qu'illustré à la figure 2.10. La courbe tracée en trait gras donne la valeur de $\bar{v}$ en fonction de x. Les coordonnées des points P, Q, R et S sont données à la page 42. On a aussi

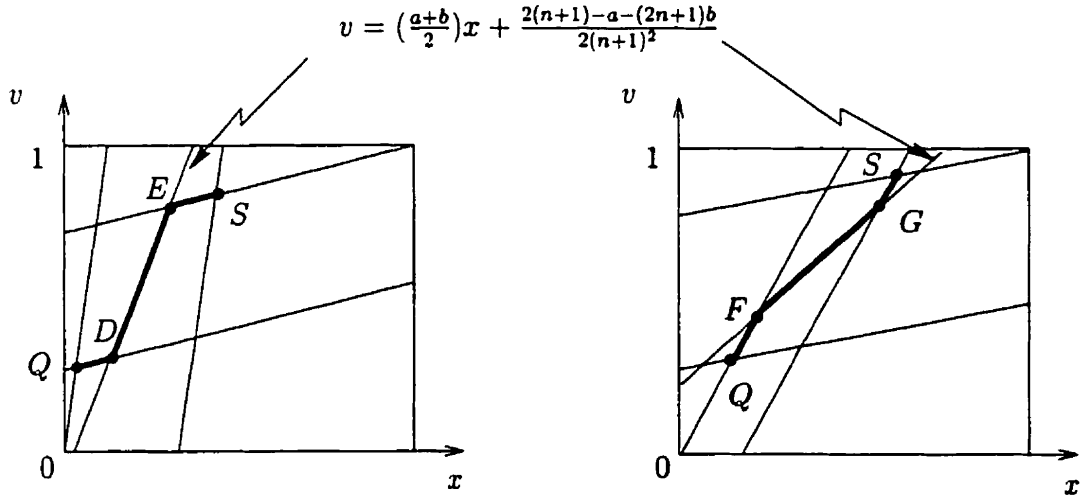
$$D = \left(\frac{1}{(n+1)^2}, \frac{n(1-b)+1}{(n+1)^2}\right)$$

$$E = \left(\frac{1}{(n+1)^2} + \frac{2n(1-b)}{(n+1)(a-b)}, \frac{b}{(n+1)^2} + \frac{2nb(1-b)}{(n+1)(a-b)} + 1 - b\right)$$

$$F = \left(\frac{2(1-b)}{(n+1)(a-b)} - \frac{1}{(n+1)^2}, \frac{2a(1-b)}{(n+1)(a-b) - \frac{a}{(n+1)^2}}\right)$$

$$G = \left(\frac{2n+1}{(n+1)^2}, \frac{n(a+1)+1}{(n+1)^2}\right).$$

Comme plus haut, nous dénotons les abscisses de ces points par l'indice x.



$(i)\ n \leq \frac{a-1}{1-b}$ $(ii)\ n \geq \frac{a-1}{1-b}$

Figure 2.10 – Deux cas pour v.

La situation (i) survient lorsque $Q_x \leq D_x$, ce qui est équivalent à $n \leq \frac{a-1}{1-b}$. Dans ce cas, il est facile de vérifier que $E_x \leq S_x$ de sorte que la droite $\mathcal{D}$ est bien telle que

représentée sur la figure. Notons que si $n = \frac{a-1}{1-b}$ alors $\mathcal{D}$ passe par Q et S. Dans le cas où $n \geq \frac{a-1}{1-b}$ la situation est telle qu'indiquée en (ii). Par la suite nous devrons pour chaque région considérer ces deux situations. Dans les deux cas (i) et (ii), l'intervalle $[Q_x, S_x]$ qui contient x se trouve divisé en trois sous-intervalles. Pour chacun de ceux-ci on détermine le maximum de ψ, qui dépend de x et que l'on dénote par $\mathcal{R}_n(x)$ (voir 2.3), puis on minimise la fonction $\mathcal{R}_n$ sur le sous-intervalle. On peut alors comparer les valeurs ainsi obtenues et déterminer le minimum, qui est la solution du problème 2.4 dans le cas où $(x, v) \in \mathrm{I}$.

Examinons d'abord la situation (i).

(1) Si $Q_x \leq x \leq D_x$ alors $\bar{v} \leq bx + \dfrac{1-b}{n+1} = v_0$ et le maximum de ψ est atteint en $v_{max} = v_0$ (figure 2.11).

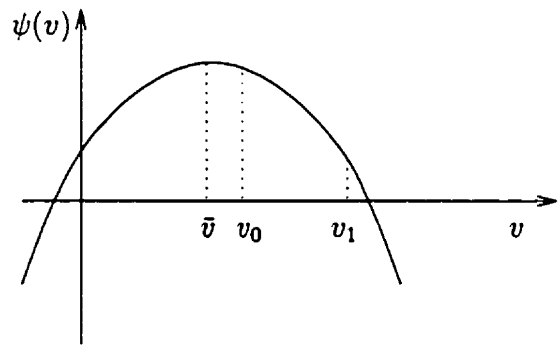


Figure 2.11 - $Q_x \leq x \leq D_x$.

On a

$$\mathcal{R}_n(x) = \frac{1}{2n^2(a-b)}\psi(v_{max}) = -\frac{1-b}{2n}x + \frac{(1-b)}{2n(n+1)}.$$

Le facteur $n^2(a-b)$ a été introduit lors de la définition de ψ et le facteur 2 provient de la définition de ϕ. Cette fonction est décroissante car $-\frac{1-b}{2n} \leq 0$ et atteint donc son

minimum en

$$x_{min} = \frac{1}{(n+1)^2}.$$

En substituant on obtient

$$\mathcal{R}_n(x_{min}) = \frac{1-b}{2(n+1)^2}.$$

(2) Si $D_x \leq x \leq E_x$ alors $\bar{v} \in [v_0, v_1]$ (voir 2.5) et $v_{max} = \bar{v}$ (figure 2.12).

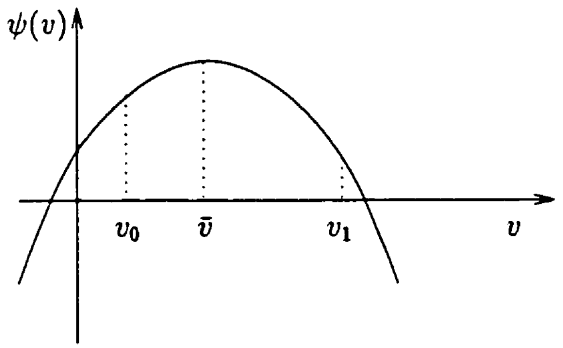


Figure 2.12 – $D_x \leq x \leq E_x$.

En substituant,

$$\mathcal{R}_n(x) = \frac{1}{2n^2(a-b)}\psi(v_{max}) = \frac{(n+1)^2(a-b)x^2}{8n^2} - \frac{(2n(1-b)+a-b)x}{4n^2}$$

$$+\frac{(a-b+4n(1-b)(n+1))}{8n^2(n+1)^2}.$$

Le minimum de cette fonction quadratique est atteint en

$$x_{min} = \frac{a-(2n+1)b+2n}{(n+1)^2(a-b)}.$$

On vérifie facilement que x_{min} est dans l'intervalle considéré. En effet,

$$\frac{1}{(n+1)^2} \leq \frac{a-(2n+1)b+2n}{(n+1)^2(a-b)}$$

$$\Leftrightarrow \quad a-b \leq a-(2n+1)b+2n$$

$$\Leftrightarrow \quad 0 \leq 1-b$$

et

$$\frac{a-(2n+1)b+2n}{(n+1)^2(a-b)} \leq \frac{a-(2n^2+2n+1)b+2n(n+1)}{(n+1)^2(a-b)}$$

$$\Leftrightarrow \quad 0 \leq 2n^2-2bn^2$$

$$\Leftrightarrow \quad 0 \leq 1-b.$$

En substituant on obtient

$$\mathcal{R}_n(x_{min}) = \frac{(1-b)(a-1)}{2(n+1)^2(a-b)}.$$

(3) Si $E_x \leq x \leq S_x$ alors $\bar{v} \geq v_1$ (voir 2.5) et le maximum de ψ est en $v_{max} = v_1 = bx+1-b$ (figure 2.13).

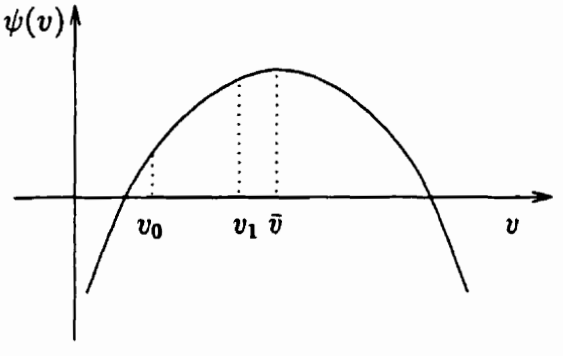


Figure 2.13 – $E_x \leq x \leq S_x$.

On a

$$\mathcal{R}_n(x) = \frac{1}{2n^2(a-b)}\psi(v_{max}) = \frac{1-b}{2}x - \frac{(1-b)^2}{2(a-b)}.$$

Cette fonction est croissante car $1-b \geq 0$ et atteint son minimum à l'extrémité gauche de l'intervalle. Après substitution, on trouve

$$\mathcal{R}_n(x_{min}) = \frac{(1-b)(a-1)}{2(n+1)^2(a-b)} + \frac{n^2(1-b)^2}{2(n+1)^2}.$$

Résumons ce que nous avons trouvé pour le cas (i). Les valeurs de $\mathcal{R}_n(x_{min})$ calculées en (1) et (3) sont supérieures à celle de (2). En effet, pour (3) ceci est clair puisque $\frac{n^2(1-b)^2}{2(n+1)^2} \geq 0$. Quant à (1), cela découle du fait que $\frac{a-1}{a-b} \leq 1$. Dans le cas (i) le minimum sur x est donc atteint en

$$\begin{aligned} x^* &= \frac{a-(2n+1)b+2n}{(n+1)^2(a-b)} \\ &= \frac{1}{(n+1)^2}\left(1+2n\frac{1-b}{a-b}\right). \end{aligned}$$

et sa valeur est égale à

$$\mathcal{R}_n(x^*) = \frac{(1-b)(a-1)}{2(n+1)^2(a-b)}.$$

Ces valeurs sont celles du théorème. Le reste de la preuve consistera à montrer que dans tous les autres cas on ne peut faire mieux.

Pour la situation de la figure 2.10(ii), les calculs sont très semblables à ceux que nous venons d'effectuer en (i). Voici un résumé succinct des résultats obtenus.

(1) Si $Q_x \leq x \leq F_x$ alors $v_{max} = ax$,

$$\mathcal{R}_n(x) = \frac{a-1}{2n^2}x + \frac{(1-b)(a-1)}{2n^2}$$

atteint son minimum à l'extrémité droite de l'intervalle et

$$\mathcal{R}_n(x_{min}) = \frac{(a-1)(1-b)}{2(a-b)(n+1)^2} + \frac{(a-1)^2}{2n^2(n+1)^2(a-b)} \geq \mathcal{R}_n(x^*).$$

(2) Si $F_x \leq x \leq G_x$ alors $v_{max} = \bar{v}$ et comme précédemment θ atteint son minimum en x^* et

$$\mathcal{R}_n(x^*) = \frac{(1-b)(a-1)}{2(n+1)^2(a-b)}.$$

(3) Si $G_x \leq x \leq S_x$ alors $v_{max} = ax + \frac{1-a}{n+1}$,

$$\mathcal{R}_n(x) = \frac{a-1}{2n}x - \frac{a-1}{2n(n+1)}$$

atteint son minimum en $x_{min} = \frac{2n+1}{(n+1)^2}$ et

$$\mathcal{R}_n(x_{min}) = \frac{a-1}{2(n+1)^2} \geq \mathcal{R}(x^*).$$

En conclusion, si $(x, v) \in$ I alors nous obtenons bien la valeur attendue pour le minimum, et celui-ci est atteint au point annoncé. On doit maintenant examiner les cas $(x, v) \in$ II, III.

Cas $(x, v) \in$ II.

Ici, $R_x \leq x \leq 1$ et $v_0 \leq v \leq v_1$ où $v_0 = x$ et $v_1 = \min\{ax + \frac{1-a}{n+1}, bx + 1 - b\}$.

Pour $(x, v) \in$ II on a trouvé plus haut (page 43) $\mu_{max} = \frac{1-v}{1-x}$. En substituant cette valeur dans l'expression de ϕ on obtient

$$\psi(v) = \phi(\mu_{max}) = -v + x + a - 1 - \frac{(1-a)^2(1-x)}{a - ax - 1 + v}.$$

Cette fonction possède une asymptote verticale en $v = ax - a + 1$ et une asymptote oblique $y = v + 1 - x - a$, tel qu'illustré à la figure 2.14.

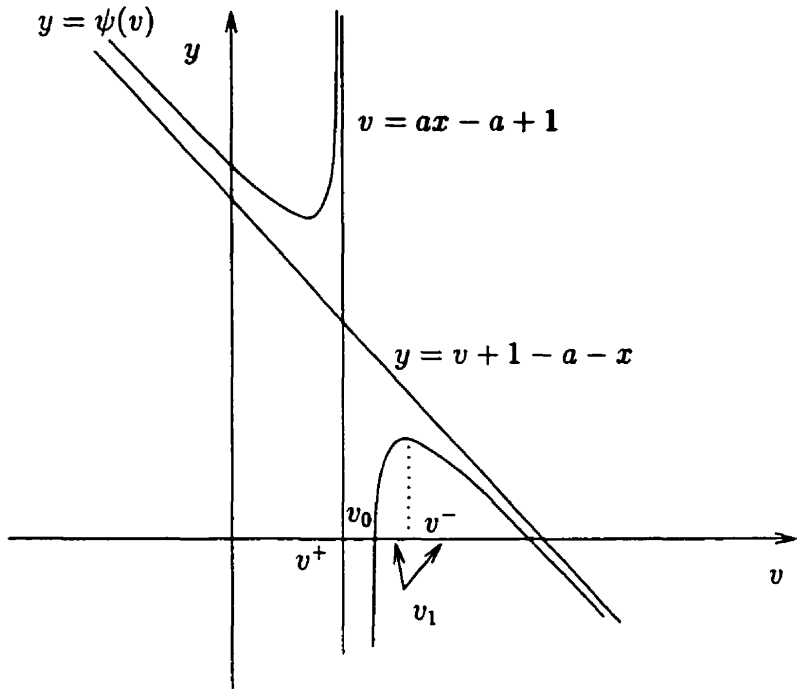


Figure 2.14 – La fonction ψ : cas $(x, v) \in$ II.

La dérivée de ψ est donnée par

$$\psi'(v) = -\frac{v^2 + 2(a - ax - 1)v + (1 - a^2 + a^2x)x}{(1 - a + ax - v)^2}$$

et les solutions de l'équation $\psi'(v) = 0$ sont

$$v^{\pm} = 1 - a + ax \pm (a - 1)\sqrt{1 - x}.$$

Le point v^- est un minimum local et v^+ un maximum local. L'asymptote verticale de ψ n'est pas dans l'intervalle $[v_0, v_1]$ car $ax - a - 1 < x \Leftrightarrow a(x-1) < x - 1 \Leftrightarrow a > 1$. Notons que $v_0 \leq v^+$ mais qu'on n'a pas nécessairement $v^+ \leq v_1$. La courbe définie par v^+ est concave puisque $(v^+)'' = -\frac{a-1}{2(1-x)^{3/2}} \leq 0$. Soit K_1 et K_2 les abscisses des

points d'intersection de cette courbe avec les droites $y = ax + \frac{1-a}{n+1}$ et $y = bx + 1 - b$ respectivement. Encore une fois, deux situations sont possibles.

(i) Si $n \leq \frac{a-1}{1-b}$ alors on peut vérifier que $K_2 \leq S_x \leq K_1$. On a donc $v^+ \leq v_1$ et $v_{max} = v_1$.

Sur $[R_x, S_x]$, $v_{max} = v_1 = ax + \frac{1-a}{n+1}$ et

$$\mathcal{R}_n(x) = \frac{1}{2}\psi(v_{max}) = \frac{a-1}{2n}\left(x - \frac{1}{n+1}\right)$$

atteint son minimum en $x_{min} = \frac{1}{n+1}$ avec $\mathcal{R}_n(x_{min}) = 0$. Cependant, du point de vue du problème originel, cette valeur n'est pas le minimum pour x dans cet intervalle. En effet, il est possible que (x, v) ne soit pas dans II mais dans I. En fait, le pire cas survient dans cette dernière région. C'est donc la valeur du minimum sur x pour $(x, v) \in$ I, calculée précédemment, qui est le minimum du pire cas lorsque $x \in [R_x, S_x]$.

Sur $[S_x, 1]$, $v_{max} = v_1 = bx + 1 - b$ et

$$\mathcal{R}_n(x) = \frac{1}{2}\psi(v_{max}) = \frac{(1-b)[(a-b)x - (1-b)]}{2(a-b)}$$

qui atteint son minimu en $x_{min} = S_x$ avec

$$\mathcal{R}_n(x) = \frac{(a-1)(1-b)}{2(n+1)(a-b)} \geq \mathcal{R}_n(x^*).$$

(ii) Si $n \geq \frac{a-1}{1-b}$ alors on vérifie que $K_1 \leq S_x \leq K_2$. On a donc $v_1 \leq v^+$ et $v_{max} = v_1$ sur $[R_x, K_1]$ et $[K_2, 1]$, ainsi que $v^+ \leq v_1$ et $v_{max} = v^+$ sur $[K_1, K_2]$.

Pour le premier de ces intervalles on a comme au cas précédent que la fonction $\mathcal{R}_n(x)$ atteint son minimum en R_x, où elle s'annule, mais ceci ne correspond pas au pire cas.

Pour le second intervalle on trouve la fonction croissante

$$\mathcal{R}_n(x) = \frac{1}{2}\psi(v_{max}) = \frac{(a-1)}{2}(1-\sqrt{1-x})^2$$

avec $x_{min} = K_1$ et

$$\mathcal{R}_n(x_{min}) = \frac{a-1}{(n+1)^2} \geq \mathcal{R}_n(x^*).$$

Pour le troisième intervalle on a comme précédemment

$$\mathcal{R}_n(x_{min}) = \frac{(a-1)(1-b)}{2(n+1)(a-b)} \geq \mathcal{R}_n(x^*).$$

En conclusion, si $(x, v) \in$ II on constate que l'on ne peut améliorer la solution déjà trouvée.

Cas $(x, v) \in$ III

Ici $0 \leq x \leq R_x$ et $v_0 \leq v \leq v_1$ où $v_0 = x$ et $v_1 = \min\{ax, bx + \frac{1-b}{n+1}\}$. On a montré plus haut (page 43) que $\mu_{max} = \frac{v}{x}$. En substituant cette valeur dans l'expression de ϕ on trouve

$$\psi(v) = n^2\phi(\mu_{max}) = -v + x - b + 1 - \frac{x(1-b)^2}{v - bx}.$$

Cette fonction possède une asymptote verticale en $v = bx$ et une asymptote oblique $y = -v + 1 - b - x$, comme sur la figure 2.15.

La dérivée de ψ est

$$\psi'(v) = \frac{v^2 - 2bxv - b^2x^2 + 2bx - b^2x - x}{(v - bx)^2}$$

et les solutions de l'équation $\psi'(v) = 0$ sont

$$v^{\pm} = bx \pm (1-b)\sqrt{x}.$$

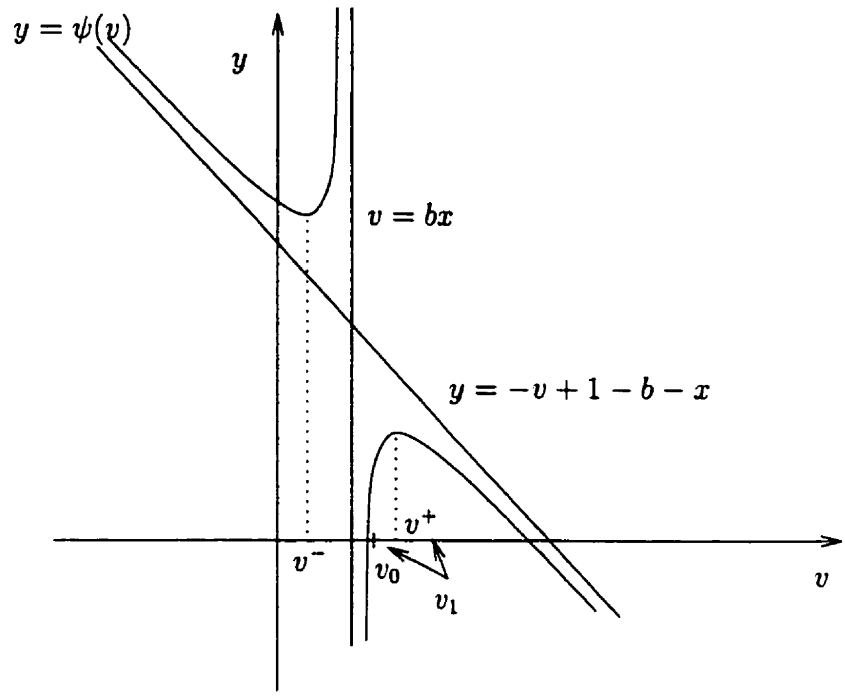


Figure 2.15 – La fonction ψ : cas $(x, v) \in$ III.

Le point v^- correspond à un maximum local de ψ et v^+ à un minimum local. Comme pour le cas précédent, l'asymptote verticale n'est pas dans l'intervalle $[v_0, v_1]$, puisque $b < 1 \Rightarrow bx < x = v_0$. On a aussi $v^- < x = v_0$. Par contre, il est possible que le maximum v^+ soit dans $[v_0, v_1]$.

La courbe définie par v^+ est concave et les abscisses de l'intersection de cette courbe avec les droites $y = ax$ et $y = bx + \frac{1-b}{n+1}$ sont $K_1 = \left(\frac{1-b}{a-b}\right)^2$ et $K_2 = \frac{1}{(n+1)^2}$ respectivement. Deux situations sont possibles (figure ??).

(i) Si $n \leq \frac{a-1}{1-b}$ alors on peut vérifier que $K_1 \leq Q_x \leq K_2$. On a donc $v^+ \geq v_1$ sur $[0, K_1]$ et $[K_2, R_x]$, et $v^+ \leq v_1$ sur $[K_1, K_2]$.

Sur le premier intervalle, $v_{max} = v_1 = ax$ et

$$\mathcal{R}_n(x) = \frac{1}{2n^2}\psi(v_{max}) = -\frac{a-1}{2n}x + \frac{(1-b)(a-1)}{2n^2(a-b)}.$$

Cette fonction est décroissante et atteint son minimum en $x_{min} = K_1$ avec

$$\mathcal{R}_n(x_{min}) = \frac{(1-b)(a-1)}{2n^2(a-b)} \geq \mathcal{R}_n(x^*).$$

Sur le deuxième intervalle, $v_{max} = v^+$ et

$$\mathcal{R}_n(x) = \frac{1}{2n^2}\psi(v_{max}) = \frac{(1-b)(1-\sqrt{x})^2}{2n^2}.$$

Cette fonction est décroissante et atteint son minimum en $x_{min} = \frac{1}{(n+1)^2}$ avec

$$\mathcal{R}_n(x_{min}) = \frac{1-b}{2(n+1)^2} \geq \mathcal{R}_n(x^*).$$

Enfin, sur le troisième intervalle, $v_{max} = v_1 = bx + \frac{1-b}{n+1}$ et

$$\mathcal{R}_n(x) = \frac{1}{2n^2}\psi(v_{max}) = \frac{(1-b)(1-(n+1)x)}{2n(n+1)}.$$

Cette fonction est décroissante et $x_{min} = \frac{1}{n+1}$ avec $\mathcal{R}_n(x_{min}) = 0$, ce qui ne correspond cependant pas au pire cas, qui survient lorsque $(x, v) \in I$.

(ii) Si $n \leq \frac{a-1}{1-b}$ alors $K_2 \leq Q_x \leq K_1$ (figure??) et on a toujours $v^+ \geq v_1$.

Sur $[0, Q_x]$, $v_{max} = v_1 = ax$ et comme pour le cas précédent sur le premier intervalle

$$\mathcal{R}_n(x_{min}) = \frac{(1-b)(a-1)}{2n^2(a-b)} \geq \mathcal{R}_n(x^*).$$

Sur $[Q_x, R_x]$, on a comme plus haut pour le troisième intervalle $\mathcal{R}_n(x_{min}) = 0$, qui ne correspond pas au pire cas.

Comme précédemment, on conclut donc que pour $(x, v) \in$ III, on n'améliore pas la solution déjà trouvée.

Ceci termine l'examen de tous les cas possibles pour (x, v). Pour compléter la preuve, il reste à traiter les cas particuliers mentionnés au début. D'abord, $a = \mu$ ne peut se produire que si $v = ax$ et alors

$$\frac{(v - x\mu)(ax - v)}{a - \mu} = 0.$$

Géométriquement, ceci correspond à un triangle OML (voir figure 2.7) d'aire nulle. De même, $b = \mu$ seulement si $v = bx + 1 - b$ et alors

$$\frac{(1 - v - b + bx)(\mu - x\mu - 1 + v)}{\mu - b} = 0.$$

Ces deux situations sont des cas particuliers du problème général.

Ensuite, puisque $A, B \geq 0$, $nA + B = 0$ ne se produit que si $A = 0$ et $B = 0$, ce qui n'est possible que pour la fonction dont le graphe est l'union des segments OP et PQ (figure 2.7) et dans ce cas, on connaît la fonction. Pour toute autre fonction, on a $nA + B > 0$. Si $nA - B = 0$ alors $\phi'(\mu)$ possède une seule solution, qui se réduit à $\mu^+ = \frac{a+b}{2}$ et le raisonnement se poursuit de la même façon, mais de beaucoup simplifié.

Enfin, $AB = 0 \Leftrightarrow A = 0$ ou $B = 0$ et on se ramène à l'un des cas traités plus haut.

Avec ces derniers détails, la preuve est complète.

2.6 Stratégie d'évaluation optimale

Nous avons mentionné à la section 2.3 que la stratégie d'évaluation gauche à droite que nous utilisons n'est pas nécessairement optimale. Dans cette section nous discutons de la formulation du problème qui résulte d'une stratégie optimale.

Il convient d'abord de clarifier ce qu'il faut entendre par une telle stratégie. Le problème est de choisir a priori, ne connaissant pas la fonction, un ordre suivant lequel les évaluations successives seront effectuées. Bien sûr, pour une fonction donnée, on pourra toujours trouver un meilleur choix, mais ceci suppose que l'on dispose d'information sur la fonction, ce qui n'est pas le cas. De plus, pour une fonction particulière, quelle que soit la stratégie employée, il sera toujours possible de trouver de meilleurs points d'évaluation, au sens où la borne sur l'erreur maximale sera moindre. Cependant ceci suppose encore une fois que l'on connaisse la fonction étudiée.

Les points d'évaluation seront donc placés séquentiellement, en minimisant chaque fois le pire cas pour l'erreur maximale, compte tenu des évaluations précédentes, et sachant combien d'évaluations il reste à faire. S'il est assez facile d'énoncer ce principe, il est est revanche plus difficile de l'exprimer par une formule mathématique, car cette formule doit contenir, implicitement du moins, le choix de l'ordre des évaluations successives. Lorsqu'il y a une seule évaluation à faire, la difficulté ne se présente pas : il y a une seule façon de placer un seul point. Dans le cas $n = 1$, le point donné par le théorème 1 est donc optimal.

Pour formuler le problème général, dénotons par $\mathcal{E}_{ij}(a,b)$ l'erreur maximale, pour x fixé, étant donné les dérivés aux extrémités a et b et sachant que i des $n-1$ points restants seront placés à gauche de x et les j autres à droite. Désignons aussi par $\mathcal{E}(n)(a,b)$ la plus petite borne supérieure sur l'erreur maximale sachant qu'il y a n évaluations à faire. On a

$$\mathcal{E}_{ij}(a,b) = \max_{v} \max_{\mu} \left\{ xv\mathcal{E}(i)\left(\tfrac{v}{x}a, \tfrac{v}{x}\mu\right) + (1-x)(1-v)\mathcal{E}(j)\left(\tfrac{1-v}{1-x}\mu, \tfrac{1-v}{1-x}b\right)\right\}.$$

Puisqu'il faut tenir compte de toutes les répartitions possibles, à gauche et à droite, des $n-1$ points restants, on doit considérer

$$\min_{i+j=n-1}\{\mathcal{E}_{ij}(a,b)\}.$$

On peut alors trouver le meilleur premier point d'évaluation en minimisant sur x :

$$\begin{aligned}\mathcal{E}(n)(a,b) &= \min_x \min_{i+j=n-1}\{\mathcal{E}_{ij}(a,b)\} \\ &= \min_{i+j=n-1} \min_x\{\mathcal{E}_{ij}(a,b)\}\end{aligned}$$

c'est-à-dire

$$\mathcal{E}(n)(a,b) = \min_{i+j=n-1} \min_x \left\{ \max_v \max_\mu \left[xv\mathcal{E}(i)\left(\tfrac{v}{x}a, \tfrac{v}{x}\mu\right) + (1-x)(1-v)\mathcal{E}(j)\left(\tfrac{1-v}{1-x}\mu, \tfrac{1-v}{1-x}b\right)\right]\right\}.$$

Avec cette formulation, on n'a pas à choisir d'avance une stratégie d'évaluation puisque l'on examine toutes les répartitions possibles à chaque fois qu'un point est ajouté. De cette façon, un certain nombre des points restants seront placés à droite et d'autres à gauche du point que l'on vient de choisir. On constate que la formulation du problème se complique et qu'il serait plus ardu de trouver une formule analogue à celle de la section 2.4. En l'absence d'une telle formule, l'évaluation de $\mathcal{E}(n)$ nécessiterait un nombre exponentiel de calculs, en raison du fait qu'il faudrait à chaque niveau de la récurrence considérer toutes les répartitions possibles des points. De plus, en pratique, il faudrait connaître à chaque itération laquelle des fonctions $\mathcal{E}_{ij}$ réalise le minimum pour pouvoir connaître la répartition à gauche et à droite des points suivants.

Il serait intéressant d'étudier plus à fond la formulation du problème pour déterminer d'une part si la borne sur l'erreur dépend du choix de la stratégie d'évaluation, et d'autre part si la borne obtenue en choisissant d'avance une stratégie est optimale.

2.7 Généralisation

Pour terminer ce chapitre, nous mentionnons la généralisation du théorème 1 à d'autres classes de fonctions. Les résultats des sections précédentes peuvent être appliqués à toute fonction f monotone dont la dérivée (variation du surgradient) est aussi monotone. En effet, dans ce cas il suffit d'une homothétie pour se ramener au cas où f est concave croissante. On peut alors calculer les points optimaux selon la formule du théorème 1 et par la transformation inverse on obtient les points optimaux pour la fonction considérée. Le tableau 2.1 expose les différents cas possibles pour des fonctions normalisées de façon analogue au cas étudié précédemment.

f	transformation
concave croissante	$(x, v) \mapsto (x, v)$
concave décroissante	$(x, v) \mapsto (1 - x, v)$
convexe croissante	$(x, v) \mapsto (1 - x, 1 - v)$
convexe décroissante	$(x, v) \mapsto (x, 1 - v)$

Tableau 2.1 – Transformations affines

CHAPITRE 3

RÉSULTATS NUMÉRIQUES

En vue d'évaluer la précision des résultats obtenus à l'aide de la formule trouvée au chapitre précédent, nous avons implanté un algorithme d'approximation de l'intégrale d'une fonction normalisée basé sur cette formule. Nous avons ensuite comparé ces résultats avec deux autres méthodes d'approximation. Nous avons également appliqué cet algorithme au problème de plus court chemin paramétrique sur deux réseaux de transport. Le détail de ces expériences numériques fait l'objet de ce chapitre, mais avant de décrire l'algorithme utilisé, quelques précisions doivent être apportées au sens à donner à la borne sur l'erreur que l'on obtient.

3.1 Le cas des fonctions linéaires par morceaux

Si f est une fonction linéaire par morceaux (*fonction LPM*) dont les points de brisure sont $\alpha_i \in [0,1], i = 1 \ldots p$, l'erreur maximale en approximant l'intégrale sera d'autant plus faible que les points d'évaluation x_k seront situés sur un plus grand nombre de sous-intervalles $[\alpha_{i-1}, \alpha_i]$, c'est-à-dire qu'il n'est pas avantageux de choisir deux points sur un même sous-intervalle. Dans le meilleur des cas, chaque sous-intervalle contiendra un point x_k et l'approximation donnera alors la valeur exacte de l'intégrale. Quelle que soit la méthode utilisée, il est donc possible, pour une fonction donnée, de choisir des points donnant une meilleure (c'est-à-dire au moins aussi

bonne) approximation. Par exemple, on pourrait choisir les points α_i pour l'approximation. Toutefois, ceci exige que l'on connaisse la fonction à évaluer, ce qui n'est pas le cas dans le problème qui nous intéresse. La formule donnée par le théorème 1 du chapitre 2 ne donne des points qu'à partir d'informations partielles sur la fonction f. De plus, cette formule minimise *le pire cas possible* étant donné l'information obtenue par l'évaluation de la fonction en certains points. Si la fonction ne représente pas le pire cas, il est possible qu'un autre choix de points donne une meilleure approximation. Il est d'ailleurs facile de trouver un exemple d'une telle situation : pour la fonction définie par les points suivants :

x	0.0	0.01	0.6	0.9	1.0
y	0.0	0.1	0.9	1.0	1.0

la borne sur l'erreur est de 14% de la valeur exacte de l'intégrale pour l'approximation avec le point optimal x^* obtenu par le théorème 1 avec $n = 1$ et de 8,8% pour celle avec le point milieu de l'intervalle [0,1]. Notons que pour cet exemple la valeur approchée de l'intégrale est la même dans les deux cas.

Avec l'augmentation du nombre n de points d'évaluation, la tendance de l'erreur est de diminuer mais il est possible, étant donnée la nature des fonctions LPM, qu'en ajoutant un point on augmente en fait la borne sur l'erreur. Il convient ici de distinguer entre la borne donnée par $\mathcal{E}_n(a,b)$, que nous appellerons *borne a priori*, qui ne dépend que de a, b et n, et la borne donnée par $\int_0^1 (U(x) - L(x))\,dx$, la *borne réelle*, qui elle dépend de la fonction et des points d'évaluation. Bien sûr $\mathcal{E}_n(a,b)$ est strictement décroissante en n, mais ce qui nous intéresse vraiment est la borne réelle et celle-ci peut parfois augmenter quand n augmente. En effet, l'ajout d'un point implique une redistribution de tous les points d'évaluation et peut donner pour une fonction particulière une moins bonne approximation. Cette situation est illustrée à la figure

3.1, où l'on voit qu'en ajoutant un point, le $(n-1)^e$ et le n^e points sont déplacés vers la droite et de ce fait le point d'intersection P ne sera plus un point de l'approximation U, mais P' le sera. En conséquence, ce qui était une représentation exacte de la fonction sur l'intervalle $[x_{i-1}, x_i]$ ne sera plus qu'une surestimation sur ce même intervalle.

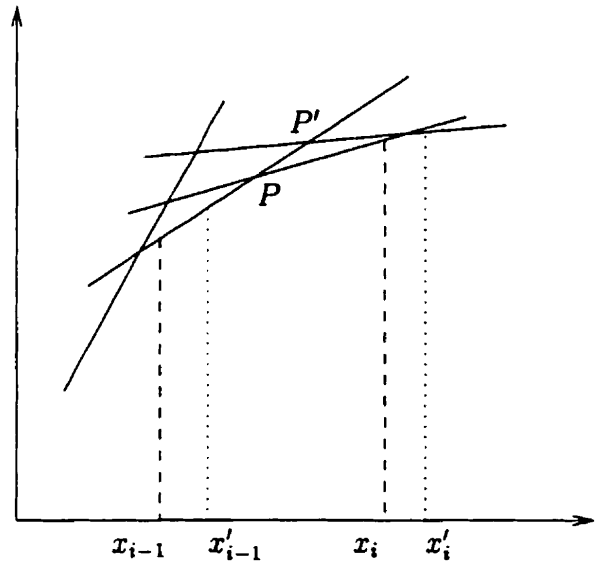


Figure 3.1 – Détérioration de l'approximation résultant de l'ajout d'un point

3.2 L'algorithme DYN

L'algorithme utilisé pour les tests est une procédure itérative très simple dont la description est donnée ci-dessous. Comme précédemment, $f'(x)$ dénote un surgradient de f en x.

Algorithme DYN

Initialisation

(0)

$$(x^0, v^0, \mu^0) := (0, f(0), f'(0))$$

$$(x^{n+1}, v^{n+1}, \mu^{n+1}) := (x_{max}, f(x_{max}), f'(x_{max}))$$

Calcul de l'approximation L

Pour $i = 1, \ldots, n$

(1)

$$a := \left(\frac{x^{n+1} - x^{i-1}}{v^{n+1} - v^{i-1}}\right) \mu^{i-1}$$

$$b := \left(\frac{x^{n+1} - x^{i-1}}{v^{n+1} - v^{i-1}}\right) \mu^{n+1}$$

(2) $$x^* := \frac{1}{(n-i+2)^2}\left(1 + 2(n-i+1)\frac{1-b}{a-b}\right)$$

(3)

$$x^i := x^{i-1} + (x^{n+1} - x^{i-1})x^*$$

$$v^i := f(x^i)$$

$$\mu^i := f'(x^i)$$

Calcul de l'approximation U

(4)
$$\begin{aligned} (\bar{x}^0, \bar{v}^0) &:= (x^0, v^0) \\ (\bar{x}^{n+1}, \bar{v}^{n+1}) &:= (x^{n+1}, v^{n+1}) \end{aligned}$$

Pour $i = 1, \ldots, n+1$

(5)
$$\begin{aligned} \bar{x}^i &:= \frac{v^{i-1} - v^i + \mu^i x^i - \mu^{i-1} x^{i-1}}{\mu^i - \mu^{i-1}} \\ \bar{v}^i &:= \mu^i (\bar{x}^i - x^i) + v^i \end{aligned}$$

Calcul de l'erreur maximale

(6)
$$\begin{aligned} \int_0^1 L(x)\, dx &:= \frac{1}{2} \sum_{i=1}^{n+1} (x^i - x^{i-1})(v^i + v^{i-1}) \\ \int_0^1 U(x)\, dx &:= \frac{1}{2} \sum_{i=1}^{n+2} (\bar{x}^i - \bar{x}^{i-1})(\bar{v}^i + \bar{v}^{i-1}) \end{aligned}$$

(7) erreur maximale $:= \int_0^1 U(x)\, dx - \int_0^1 L(x)\, dx$

À l'étape 1 on normalise les dérivées aux extrémités en fonction des points (x^{i-1}, v^{i-1}) et (x_{max}, v_{max}). Si f est une fonction normalisée, ce dernier point est simplement (1,1). Le point optimal x^* est calculé à l'étape 2, puis une mise à l'échelle est effectuée en 3 pour donner le nouveau point x^i et permettre d'évaluer v^i et μ^i. L'étape 4 reprend la formule déjà donnée à la section 2.2 pour le calcul de l'approximation U. Les intégrales de L et S sont calculées à l'étape 5. Finalement, à l'étape 6, l'erreur maximale est calculée, tel que décrit à la section 2.3.

L'algorithme ci-dessus présente l'inconvénient que le nombre de points d'évaluation doit être décidé d'avance et qu'on ne peut le modifier en cours d'exécution. En effet, la position de chaque point est fonction du nombre total de points n et si on change ce nombre, par exemple pour obtenir une approximation plus précise en faisant une évaluation de plus, on doit changer tous les points, c'est-à-dire reprendre tout le calcul. Cette limitation est inhérente à la programmation dynamique. Elle apparaît également dans la méthode Fibonnacci pour trouver l'extrémum d'une fonction unimodale (voir [1]). Pour y pallier, nous proposons la méthode suivante. Puisqu'on cherche à obtenir une borne supérieure sur l'erreur, on peut d'abord estimer en calculant $\mathcal{E}_n(a,b)$ le nombre de points maximum nécessaires pour une approximation avec une erreur inférieure à une tolérance donnée, disons ϵ. Comme nous le verrons plus tard, cette borne a priori est souvent beaucoup moins bonne que la borne réelle puisqu'elle est calculée à partir des seules valeurs a et b. On peut cependant, à chaque itération, comparer l'erreur maximale pour les points déjà trouvés avec ϵ et s'arrêter aussitôt qu'on se trouve en-deçà. Ceci peut se faire de façon efficace et en ayant soin de choisir n tel que $\mathcal{E}_n(a,b) < \epsilon$ on s'assurera d'une erreur inférieure à ϵ en ayant possiblement à effectuer moins de n évaluations. Ceci est illustré par les tableaux 3.3 et 3.4 de la section 3.7.

3.3 Heuristiques d'approximation

Nous avons comparé les résultats obtenus avec l'algorithme décrit à la section précédente avec ceux de deux méthodes heuristiques, que nous décrivons ici.

3.3.1 L'heuristique UNI

La méthode UNI consiste simplement à choisir des points uniformément distribués sur l'intervalle considéré, c'est-à-dire

$$x^i = c + \frac{i}{n}(d - c)$$

si l'on a n points d'évaluation sur l'intervalle $[c, d]$. Il est à noter que cette méthode diffère de DYN en ce que les points d'évaluation sont déterminés à l'avance et que l'information recueillie lors des évaluations n'est pas mise à profit. C'est donc une méthode passive.

3.3.2 L'heuristique INTER

La deuxième méthode, INTER, est une version de l'algorithme du sandwich tel que décrit à la section 1.3.3. La différence avec les algorithmes proposés dans les travaux cités dans cette section est dans la façon de choisir le point qui subdivise le sous-intervalle présentant la plus grande erreur : ici nous utilisons le point donné par le théorème 1, qui minimise l'erreur dans le pire cas sur le sous-intervalle. De façon plus générale, INTER subdivise le sous-intervalle d'erreur maximale à l'aide de m points donnés par le théorème 1.

Heuristique INTER

Initialisation

Soit ϵ une tolérance sur l'erreur.

erreur initiale : $E := \infty$.

$$(0) \qquad \begin{array}{lcl} (x^0, v^0, \mu^0) & := & (0, f(0), f'(0)) \\ (x^{n+1}, v^{n+1}, \mu^{n+1}) & := & (x_{max}, f(x_{max}), f'(x_{max})) \end{array}$$

Soit $[c, d] := [0, 1]$ l'intervalle courant.

Calcul de l'approximation L

Tant que $E \geq \epsilon$

(1) Subdiviser $[c, d]$ à l'aide de la méthode DYN avec m points.

(2) Calculer l'erreur a priori sur chacun des sous-intervalles pour les points déjà trouvés. Soit $[c, d]$ le sous-intervalle présentant la plus grande erreur.

Calcul de l'approximation U

(4) Calculer les point $(\bar{x}^i, \bar{v}^i)$ à l'aide des points (x^i, v^i).

Calcul de l'erreur maximale

(6) Calculer l'intégrale de L et de U.

(7) Calculer l'erreur maximale : erreur $:= \int_0^1 (U - L)$.

Notons qu'avec INTER le nombre de points d'évaluation n'a pas à être déterminé d'avance : on ajoute les points jusqu'à ce que l'on obtienne la précision voulue.

3.4 Méthodologie des tests

Dans le but d'évaluer l'efficacité de la méthode DYN, nous avons effectué deux ensembles de tests. Nous avons d'abord comparé la borne réelle sur l'erreur obtenue avec DYN, UNI et INTER. Cette borne, rappelons-le, est donnée par $\int_0^1 (U - L)$. Les fonctions utilisées étant connues, nous avons exprimé l'erreur comme un pourcentage de la valeur réelle de l'intégrale. Nous avons ensuite comparé la borne a priori $\mathcal{E}_n(a, b)$ avec la borne réelle obtenue après évaluation, ainsi qu'avec l'erreur réelle $\int_0^1 (U - f)$ commise en utilisant l'approximation de U de f.

Les comparaisons ont porté sur des fonctions concaves croissantes normalisées, lisses d'une part et linéaires par morceaux d'autre part. Pour le cas lisse, la fonction suivante a été utilisée :

$$f(x) = (b-1)(1-x)^{\frac{a-b}{1-b}} + bx + 1 - b.$$

On peut vérifier facilement que f est concave croissante sur [0,1] et satisfait $f(0) = 0$, $f(1) = 1$, $f'(0) = a$ et $f'(1) = b$. Quatre classes de fonctions ont été retenues, correspondant à différents rapports entre a et b, tel qu'illustré au tableaux 3.1 et 3.2. Les fonctions LPM ont été obtenues à partir de f en évaluant en 60 points choisis aléatoirement sur [0,1].

Les fonctions utilisées pour les tests se répartissent en quatre types, selon leurs dérivées (surgradients) a et b en 0 et 1. Rappelons que $1 \leq a < \infty$ et que $0 \leq b \leq 1$. Dans le tableau 3.1, *a faible* signifie donc que a est proche de 1 et *b forte* que b est proche de 1.

Les fonctions de types I et III présentent une plus forte courbure, ou variation du surgradient dans le cas LPM, tandis que celles de types II et IV ont une courbure faible. Les valeurs de a et b utilisées sont données au tableau 3.2 et le graphe des fonc-

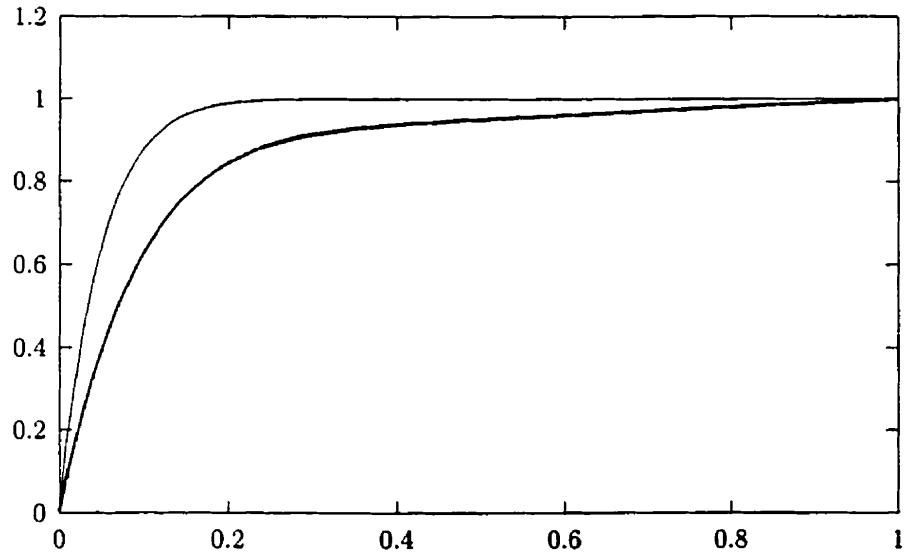
tions aux figures 3.2 et 3.3. Seules les fonctions lisses sont représentées, les fonctions LPM ayant une forme très similaire.

Tableau 3.1 – Types de fonctions

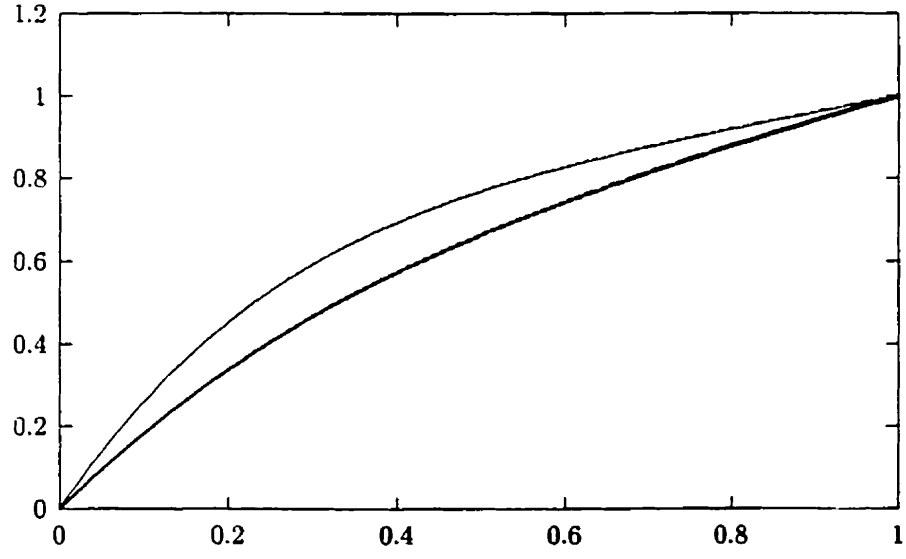
Type	a	b
I	forte	faible
II	faible	forte
III	forte	forte
IV	faible	faible

Tableau 3.2 – Fonctions test : paramètres

Type	a	b
I	20	0
	10	0.1
II	3	0.4
	2	0.6
III	20	0.4
	10	0.6
IV	2	0
	3	0.1

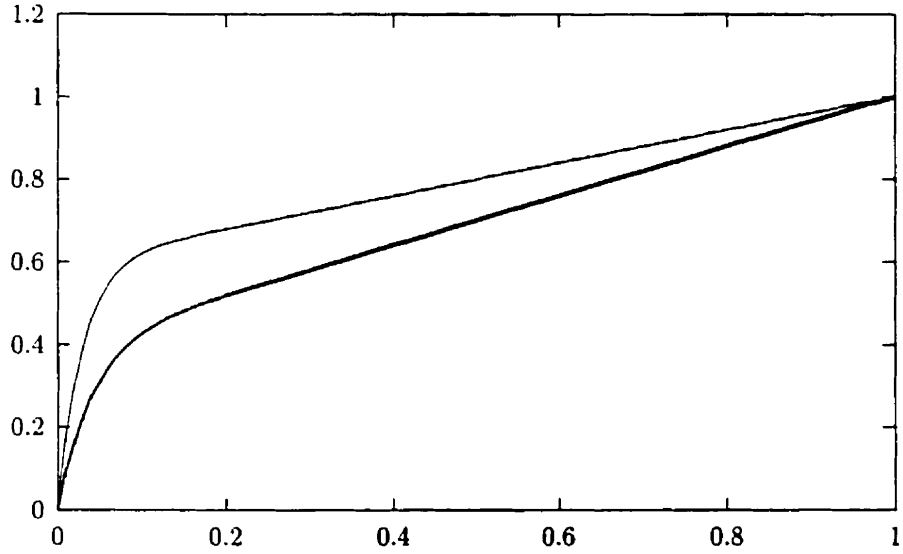


Type I

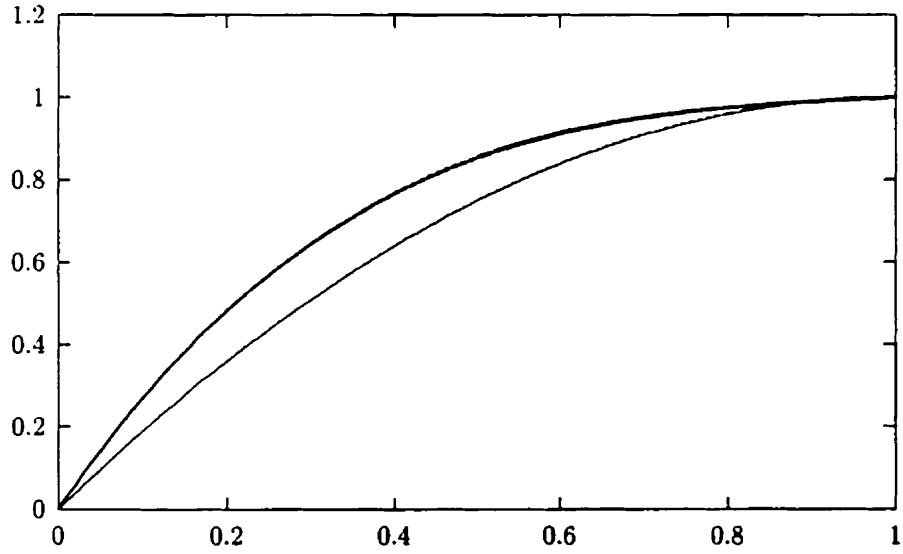


Type II

Figure 3.2 – Fonctions test lisses : types I et II



Type III



Type IV

Figure 3.3 – Fonctions test lisses : types III et IV

3.5 Comparaison de DYN avec les méthodes heuristiques

Nous présentons dans cette section les résultats des comparaisons de la méthode DYN avec les heuristiques décrites en 3.3.

3.5.1 Comparaison de DYN, INTER et UNI

Les figures 3.4 à 3.7 (cas lisse) et 3.8 à 3.11 (cas LPM) montrent les résultats obtenus lors de la comparaison des méthodes DYN, INTER et UNI. En abscisse se trouve le nombre de points d'évaluation, excluant l'évaluation aux extrémités. En ordonnée, l'erreur maximale est exprimée de façon relative en divisant l'erreur $\int_0^1 (U - L)$ par la valeur réelle de l'intégrale $\int_0^1 f$ puis en multipliant par 100 pour obtenir un pourcentage. Nous avons utilisé une échelle logarithmique pour l'axe des ordonnées.

On constate d'abord que DYN est toujours meilleure que UNI et que lorsque les dérivées en 0 ou en 1, sont proches de 1 les trois méthodes donnent des résultats semblables. Ceci était prévisible puisque plus on s'approche du graphe de $y = x$, moins l'erreur est importante.

La méthode INTER s'avère meilleure que DYN pour les fonctions dont la courbure est forte près de l'extrémité gauche de l'intervalle [0,1] et ce de façon plus marquée pour les fonctions LPM. Ceci reflète le fait que INTER peut mieux s'adapter à cette situation que DYN, qui, une fois placé un point, doit placer le suivant à droite de celui-ci. On constate cependant que DYN donne de meilleurs résultats lorsque la variation de la dérivée est plus uniforme sur [0,1]. Dans ce cas INTER est même parfois moins

efficace que UNI. On remarque enfin que INTER présente une diminution brusque de l'erreur lorsque le nombre de points d'évaluation est de la forme $2^p - 1$. Ceci s'explique comme suit. Si $2^p - 1$ points ont été placés, on a 2^p sous-intervalles. Tant que l'on n'a pas ajouté 2^p nouveaux points, au moins un de ceux-ci n'a pas été subdivisé et présente une «grande» erreur. Lorsque 2^p points sont ajoutés, pour un total de $2^{p+1} - 1$, les erreurs sur les sous-intervalles tendent à être semblables et l'erreur diminue rapidement.

En conclusion, la comparaison des méthodes DYN et INTER ne montre pas que l'une est systématiquement meilleure que l'autre. De plus, ces deux méthodes sont simples à implanter et sont semblables du point de vue du temps de calcul. On pourra donc utiliser l'une ou l'autre, le choix dépendant du problème et de la fonction à approximer.

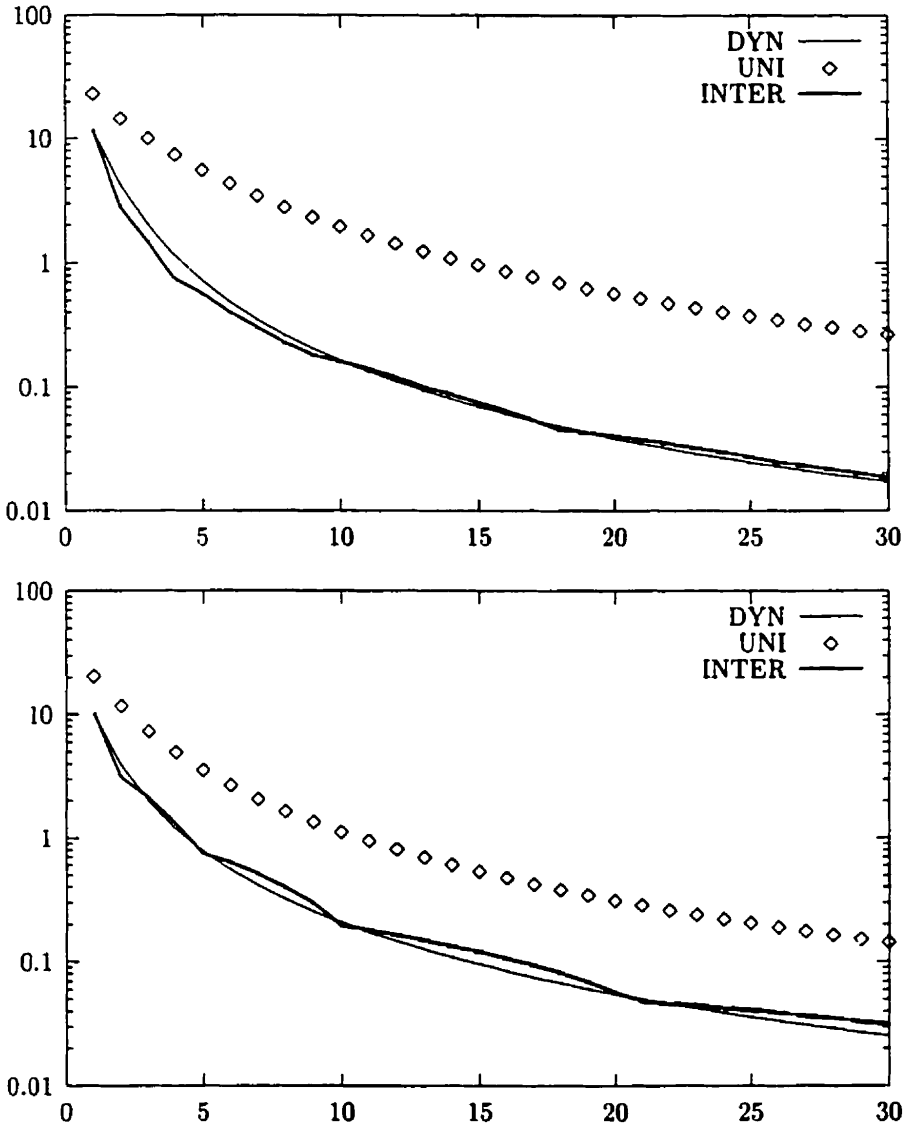


Figure 3.4 – Erreur maximum : fonctions lisses de type I

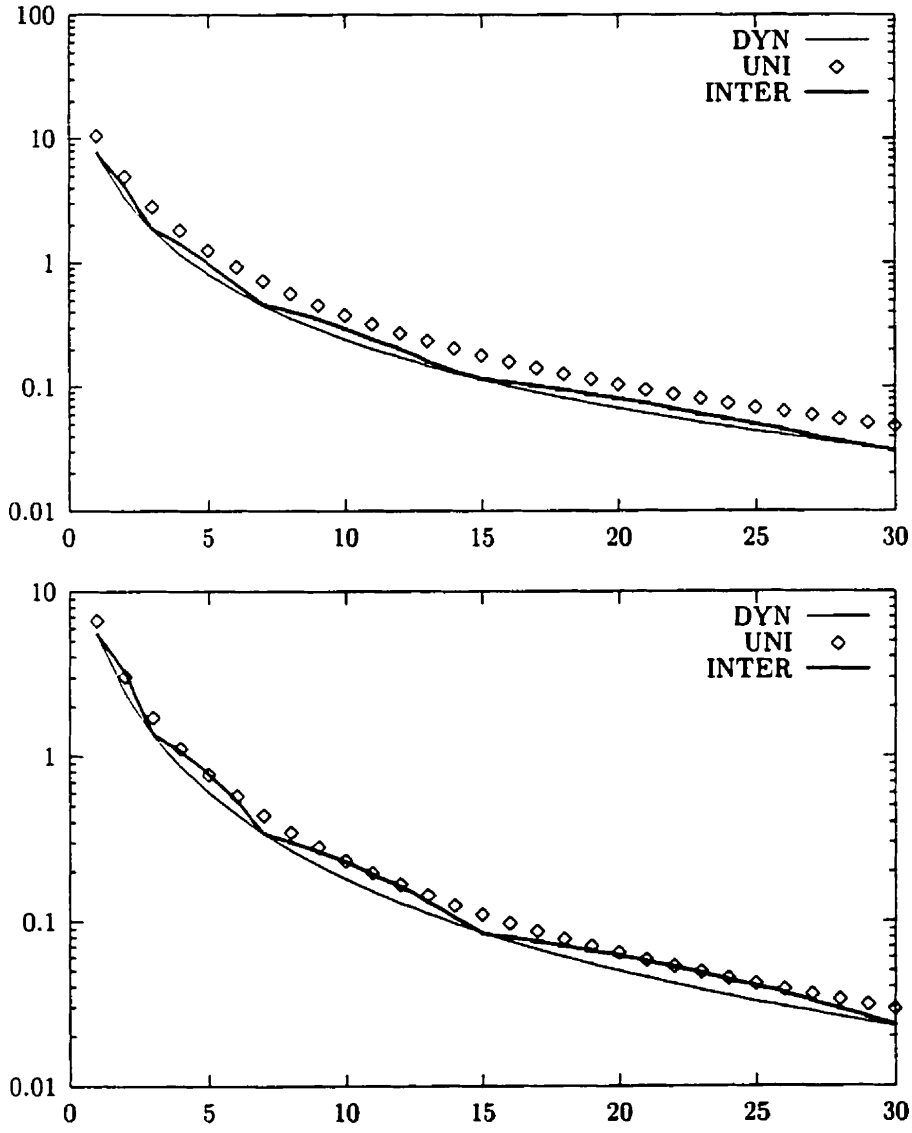


Figure 3.5 – Erreur maximum : fonctions lisses de type II

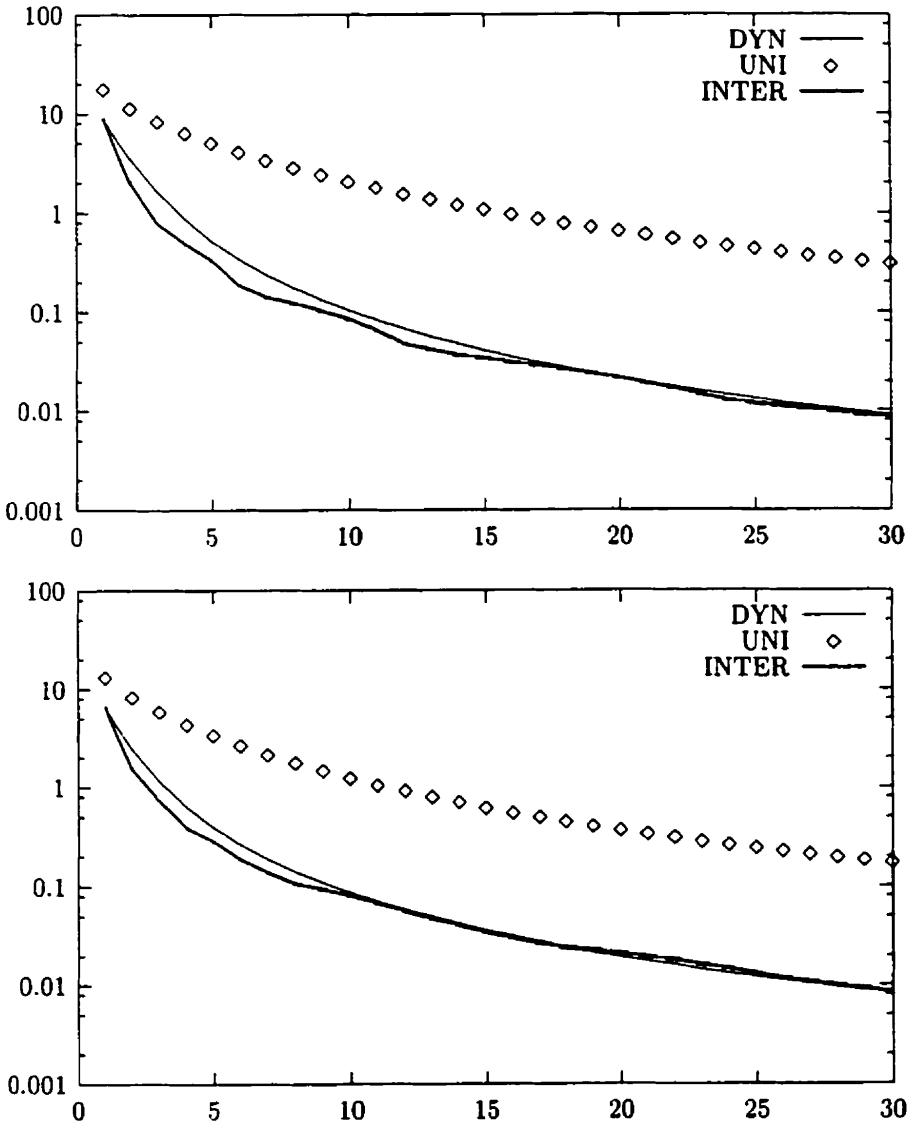


Figure 3.6 – Erreur maximum : fonctions lisses de type III

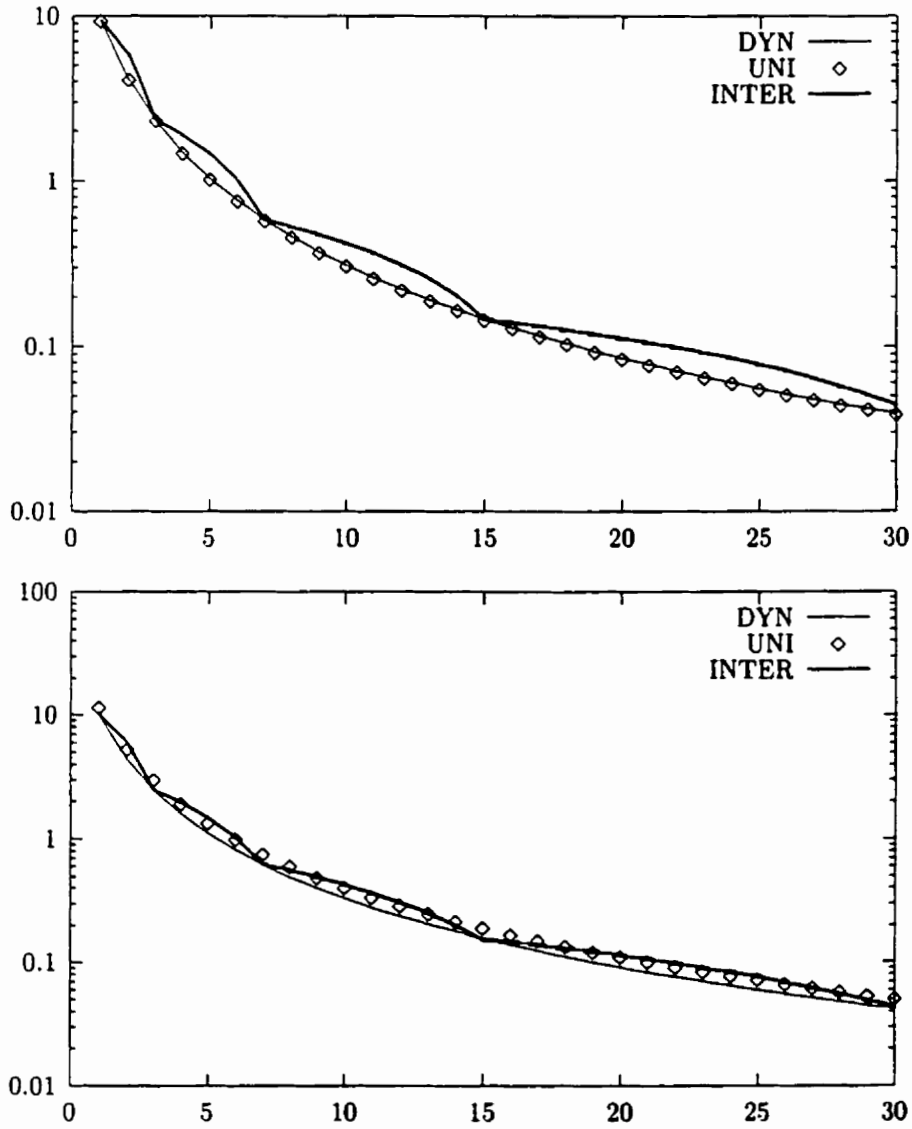


Figure 3.7 – Erreur maximum : fonctions lisses de type IV

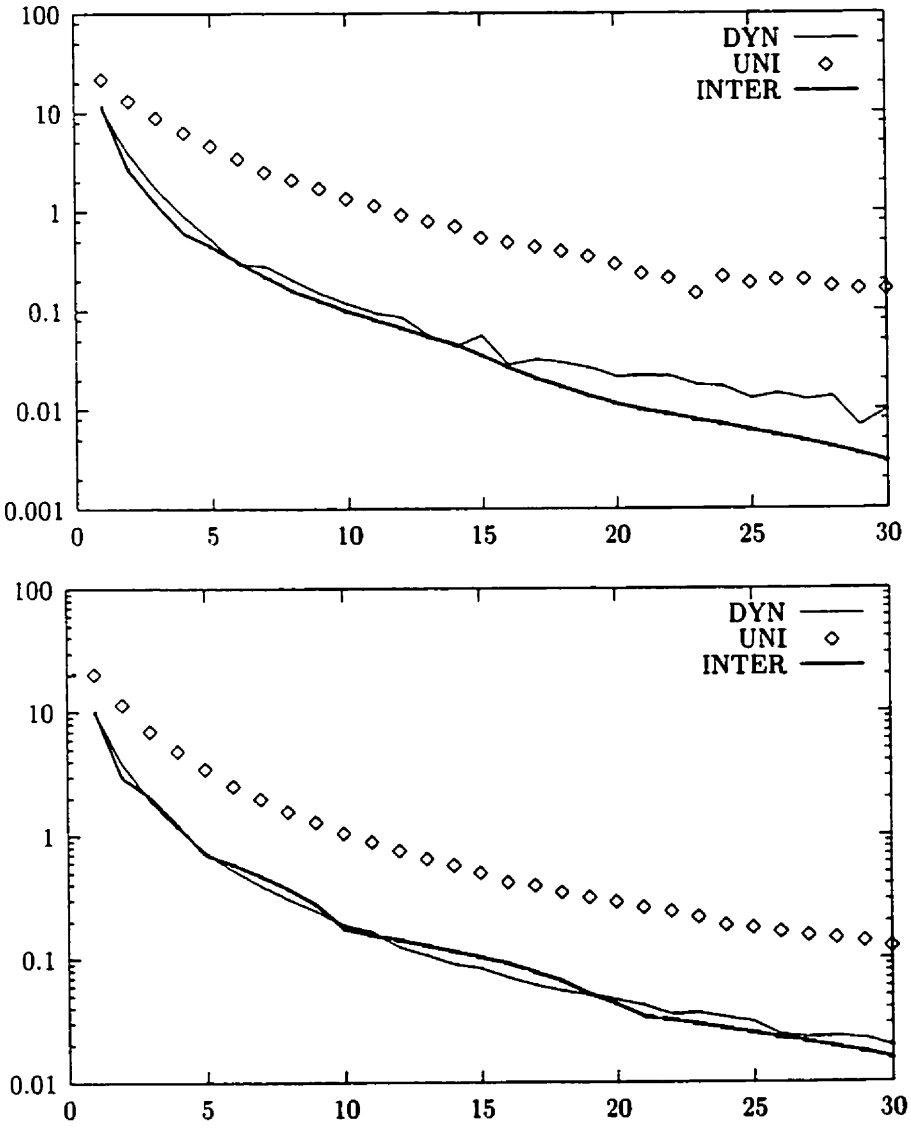


Figure 3.8 – Erreur maximum : fonctions LPM de type I

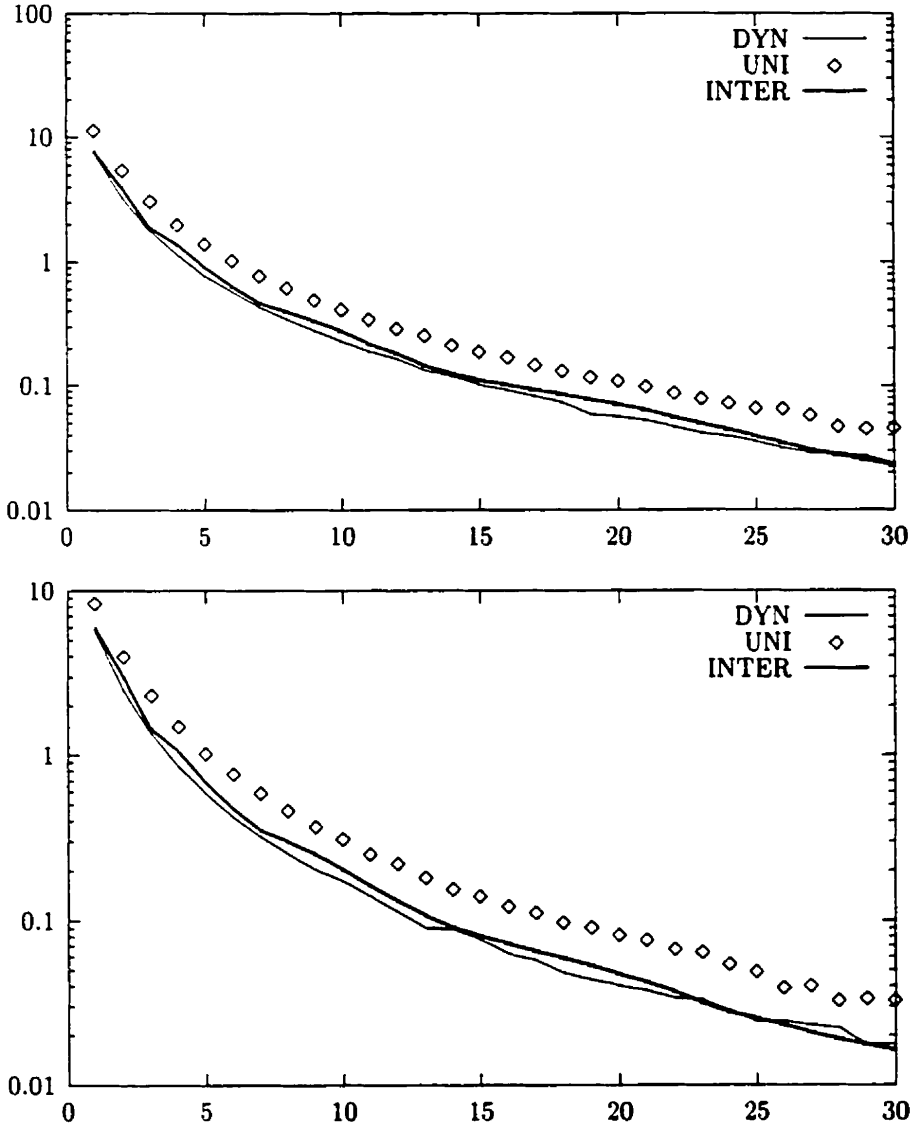


Figure 3.9 – Erreur maximum : fonctions LPM de type II

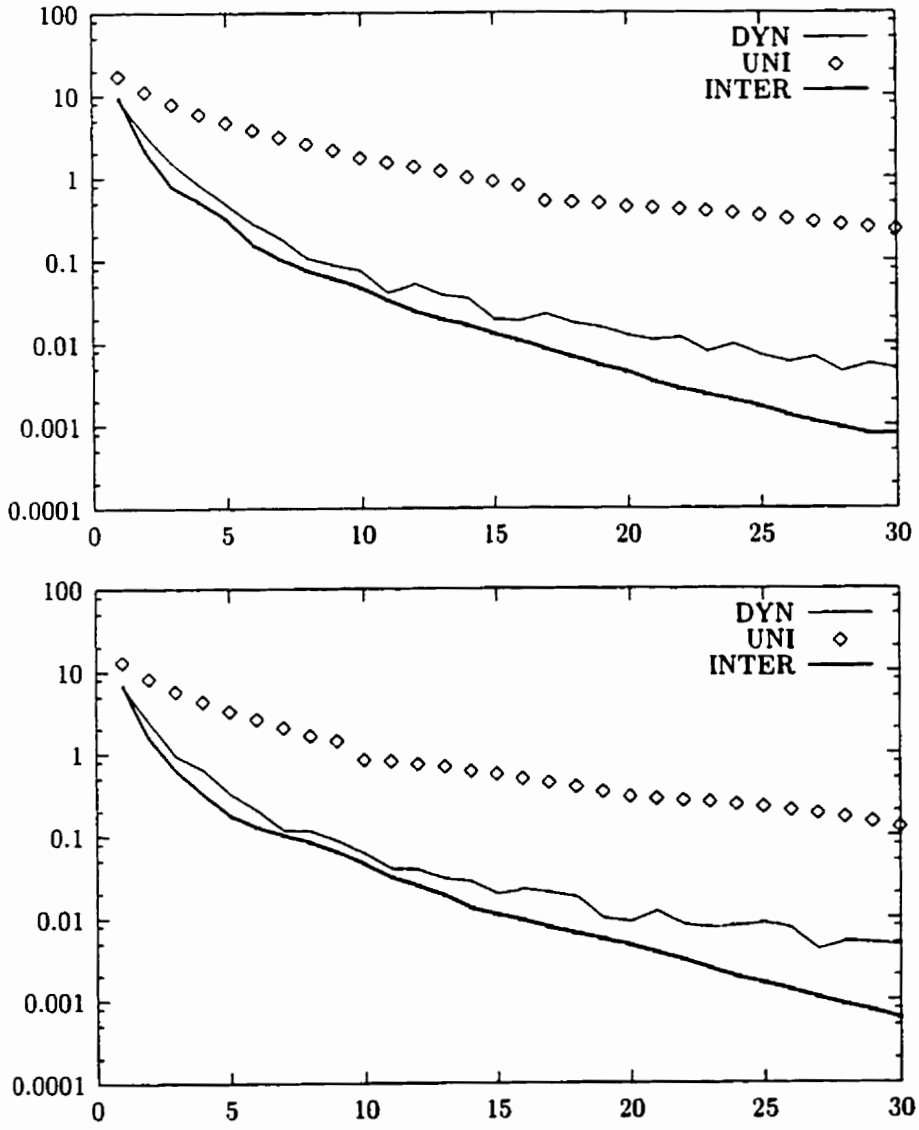


Figure 3.10 – Erreur maximum : fonctions LPM de type III

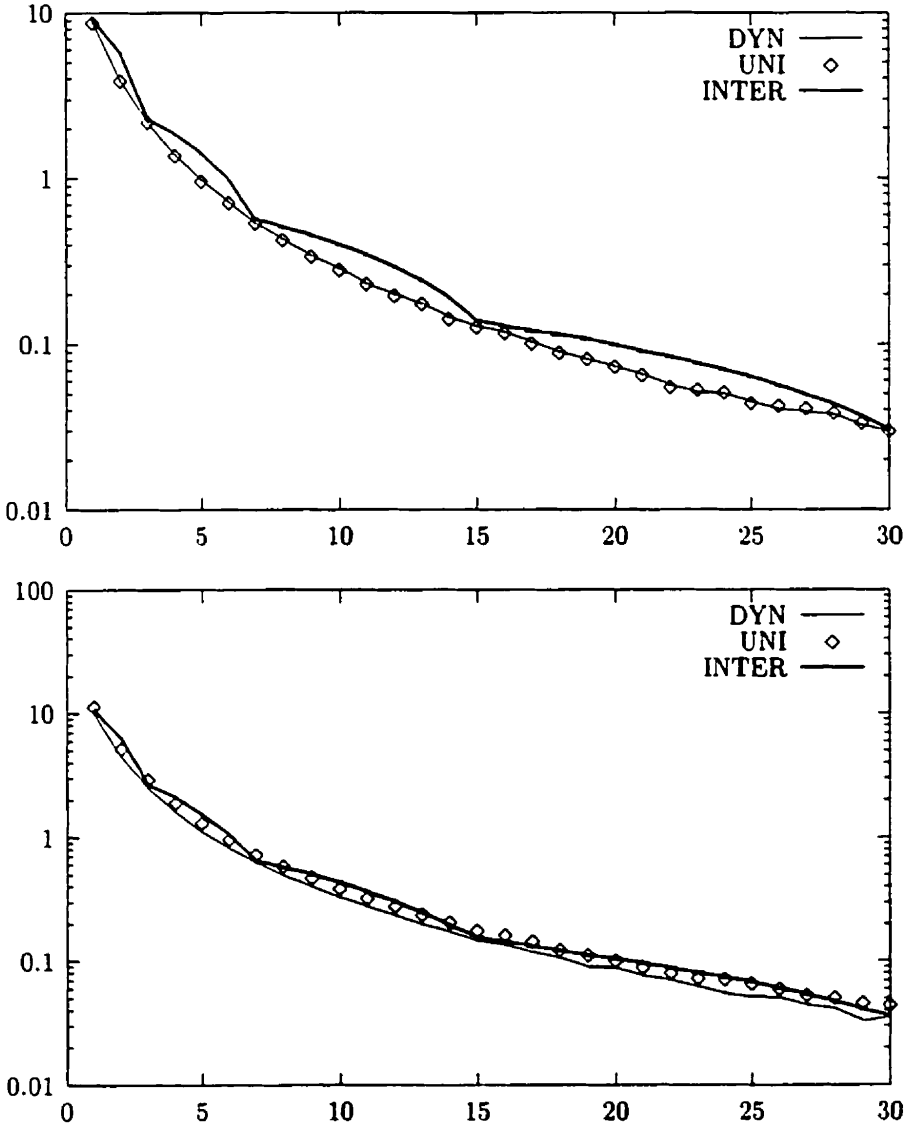


Figure 3.11 – Erreur maximum : fonctions LPM de type IV

3.6 Comparaison des bornes a priori et réelles pour la méthode DYN

Les figures 3.12 à 3.15 montrent les résultats de la comparaison de la borne a priori $\mathcal{E}_n(a,b)$ et de la borne réelle, ainsi que la valeur réelle de l'erreur. Comme auparavant, les erreurs sont exprimées comme un pourcentage de la valeur exacte de l'intégrale. Le nombre de points d'évaluation est en abscisse et l'échelle de l'ordonnée est logarithmique.

La borne réelle est évidemment toujours meilleure que la borne a priori, mais on constate que l'écart entre celle-ci et l'erreur réelle est parfois important, surtout quand les dérivées aux extrémités s'éloignent de 1 (la courbure de la fonction est forte). Ceci donne à penser qu'il y aurait possibilité d'améliorer la borne sur l'erreur, possiblement en développant de nouvelles heuristiques.

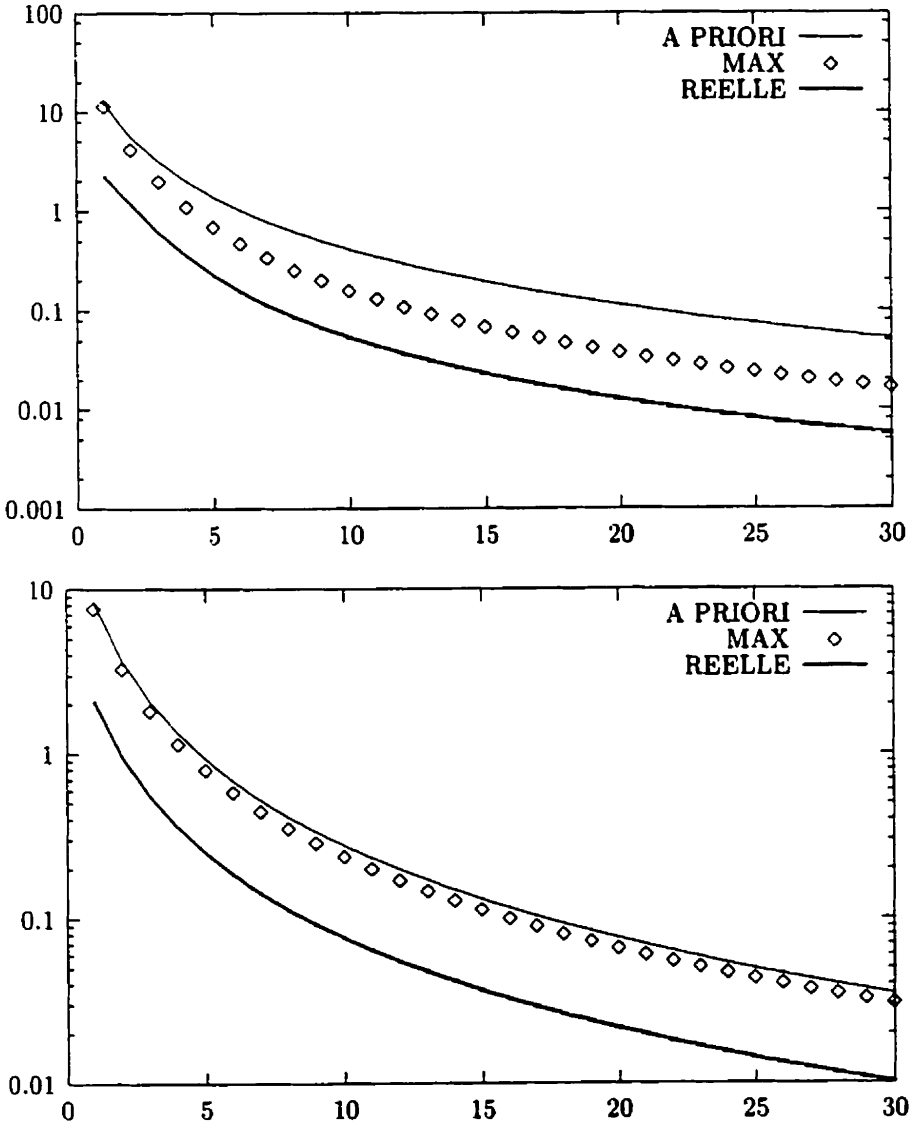


Figure 3.12 – Méthode DYN : fonctions lisses de type I et II

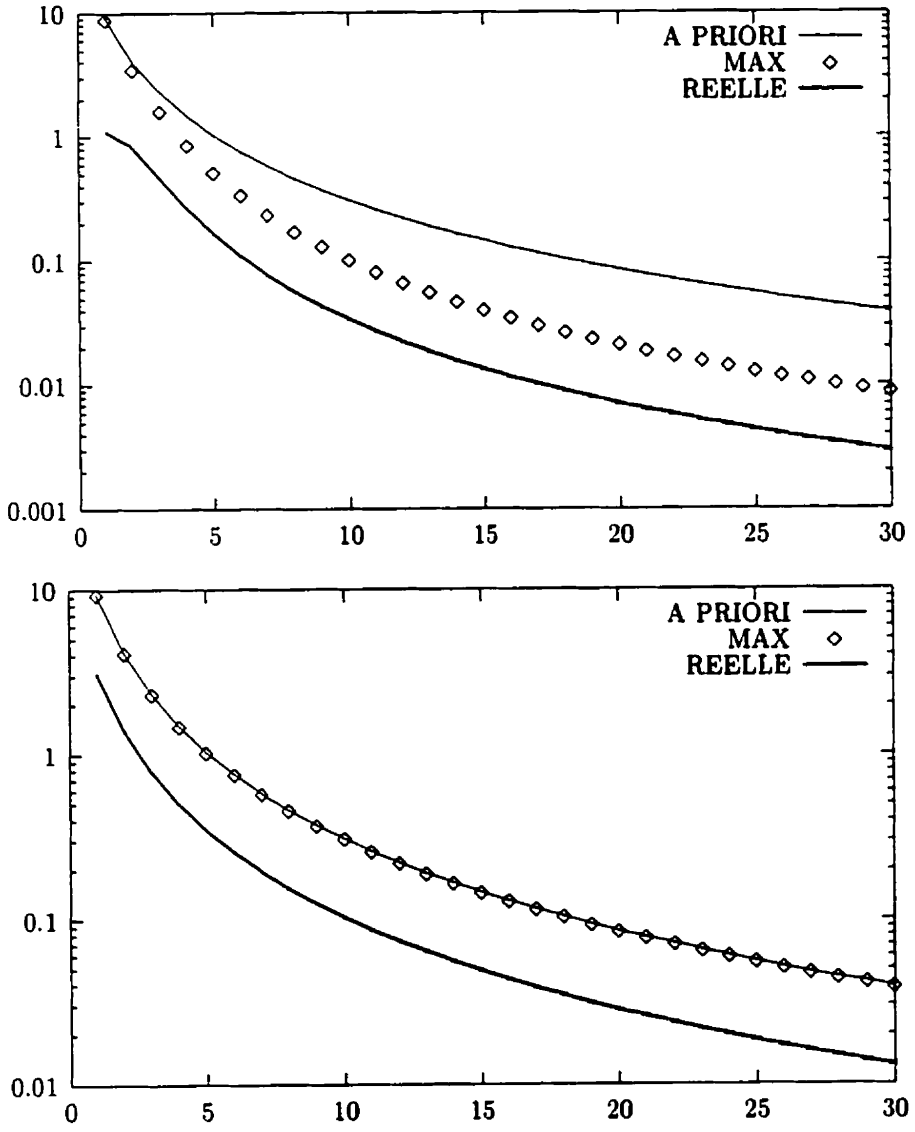


Figure 3.13 – Méthode DYN : fonctions lisses de type III et IV

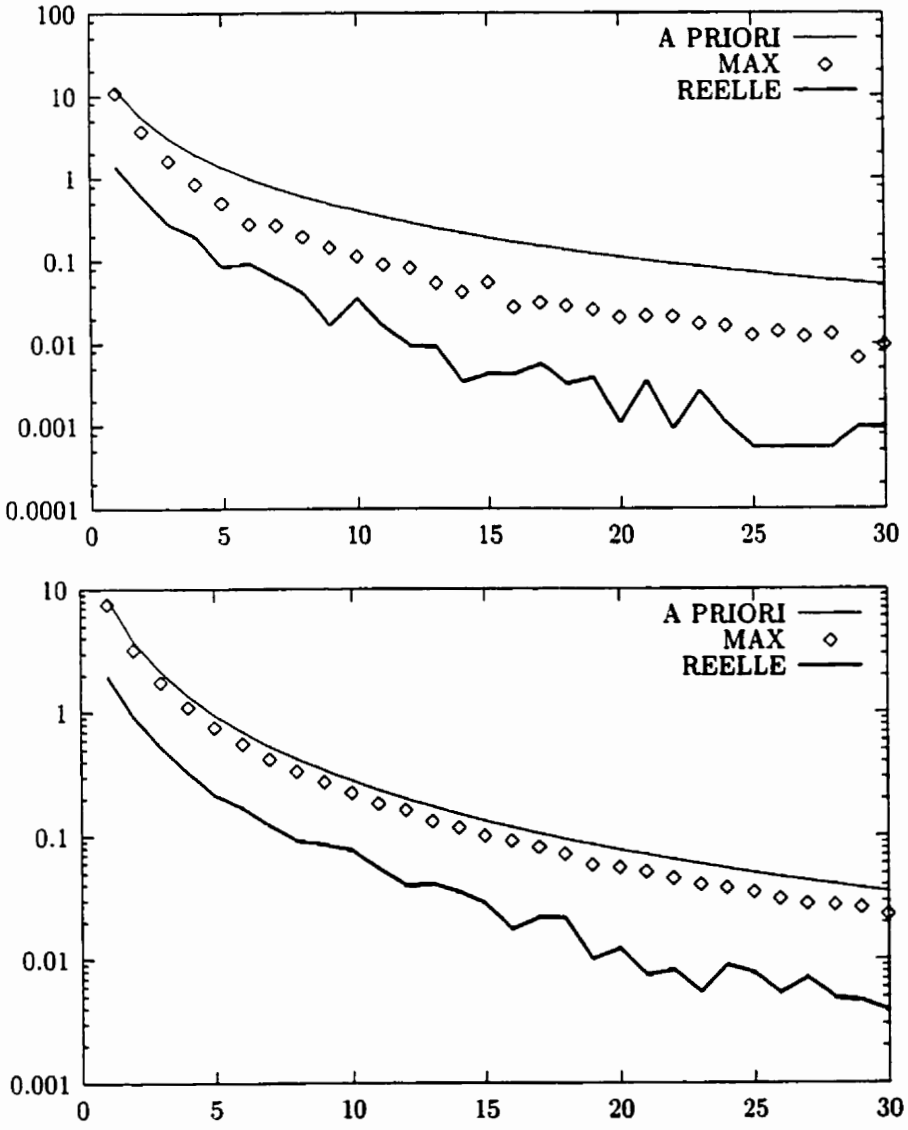


Figure 3.14 – Méthode DYN : fonctions LPM de type I et II

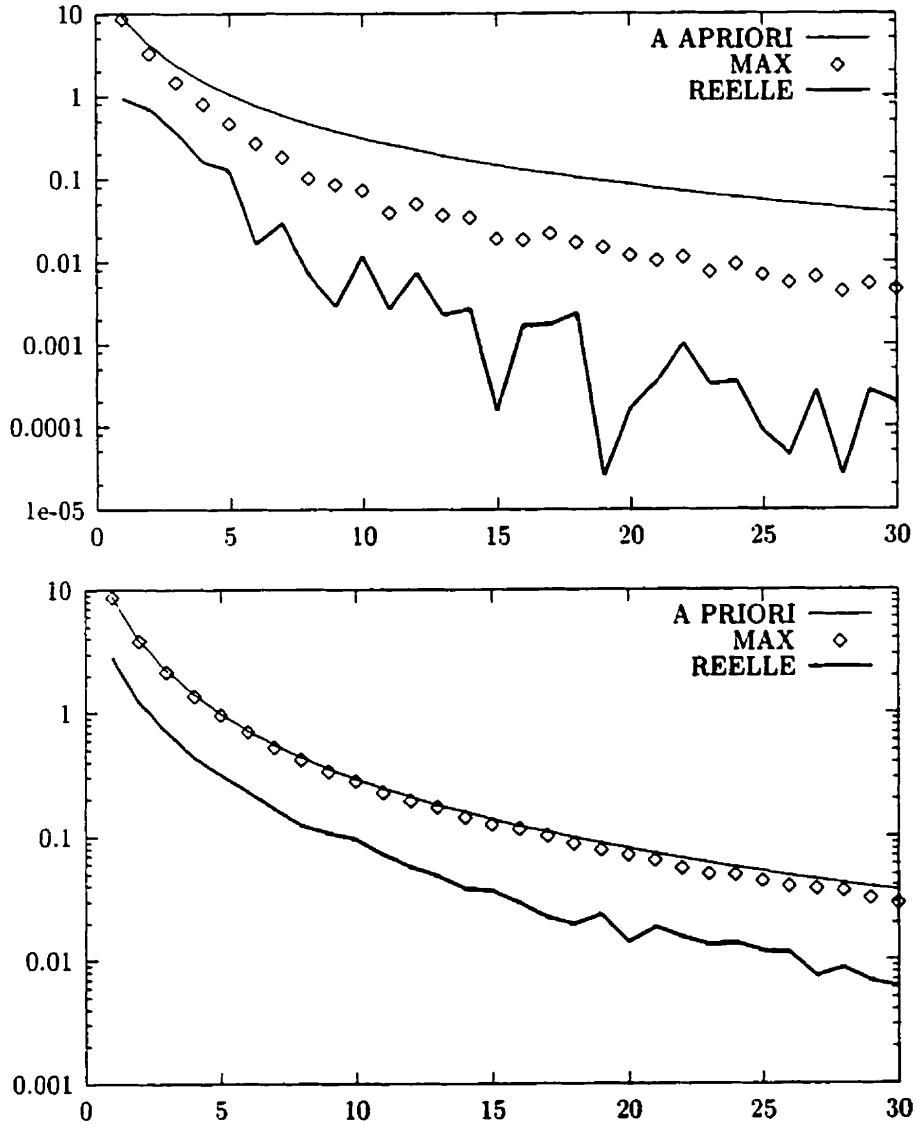


Figure 3.15 – Méthode DYN : fonctions LPM de type III et IV

3.7 Nombre de points nécessaires

Les tableaux 3.3 et 3.4 donnent pour quelques valeurs de ϵ une borne N sur le nombre de points d'évaluation nécessaire, pour obtenir une erreur inférieure à ϵ, ainsi que le nombre n de points requis en pratique pour obtenir une telle précision. Nous considérons ici l'erreur absolue $\int_0^1 (U - L)$ (et non plus l'erreur relative comme dans les sections précédentes). Dans tous les cas l'erreur est bornée par ϵ mais on peut parfois réduire le nombre d'évaluations, comme expliqué à la section 3.2. Encore une fois, c'est dans le cas des fonctions ayant une forte courbure que la méthode est la plus efficace.

Tableau 3.3 – Cas lisse.

		$\epsilon = 0.01$		$\epsilon = 0.001$		$\epsilon = 0.0001$	
a	b	N	n	N	n	N	n
20.00	0.00	6	4	21	18	68	65
10.00	0.10	6	5	20	19	63	61
3.00	0.40	4	4	15	15	48	48
2.00	0.60	3	3	11	11	37	37
20.00	0.40	5	3	17	13	53	48
10.00	0.60	4	3	13	10	43	39
2.00	0.00	4	4	15	15	49	49
3.00	0.10	5	5	17	17	55	55

Tableau 3.4 – Cas LPM.

		$\epsilon = 0.01$		$\epsilon = 0.001$		$\epsilon = 0.0001$	
a	b	N	n	N	n	N	n
13.98	0.00	6	4	21	18	68	65
9.47	0.10	6	5	20	19	63	61
2.32	0.40	4	4	15	15	48	47
2.58	0.60	3	3	12	12	39	39
16.14	0.40	5	3	16	12	53	47
9.35	0.60	4	2	13	10	43	39
1.96	0.05	4	4	15	15	48	47
3.08	0.10	5	5	17	17	55	54

3.8 Tests avec la fonction objectif du problème d'équilibre bicritère

Comme mentionné au chapitre 1, la motivation principale de ce travail était de trouver une bonne solution approchée au problème de plus court chemin paramétrique (le problème PCCP) qui intervient lors de la résolution du problème d'équilibre bicritère. Nous avons donc adapté l'algorithme DYN à ce problème. La seule modification nécessaire pour appliquer DYN à la fonction objectif du problème de PCCP consiste à remplacer l'expression de la fonction f, aux étapes 0 et 3, par un algorithme de plus court chemin. Cependant en pratique la valeur de la fonction objectif est moins importante que la solution optimale du problème, c'est-à-dire la donnée du flot sur chaque arc du réseau. C'est pourquoi nous avons inclus dans notre algorithme modifié le calcul de ces flots, même si cette information n'est pas nécessaire aux résultats que

nous présentons dans cette section. En raison de la taille des réseaux, les flots sont calculés lors des évaluation mais ne sont pas stockés en mémoire. En conséquence, on doit apporter une autre modification à l'algorithme DYN pour permettre de calculer les points de l'approximation U au fur et à mesure plutôt qu'après avoir trouvé tous les points x^i. Ceci se fait aisément : il suffit d'inclure l'étape 4 de l'algorithme DYN de la section 3.2 dans la boucle qui calcule les points x^i, soit immédiatement après l'étape 3.

Nos tests ont porté sur deux réseaux : l'un de petite taille, celui de Sioux Falls, et l'autre de grande taille, celui de Montréal. Le coût sur les arcs est de la forme $F+\alpha G$, où α joue le rôle tenu par la variable x dans les sections précédentes. Pour chaque arc les constantes F et G ont été choisies aléatoirement sur un intervalle prédéterminé de façon à ce que F soit toujours strictement positif et que G soit non nul sur 20% des arcs.

En plus des coûts sur les arcs, un ensemble de paires origine-destination (*paires O-D*) est nécessaire. Pour chacune de ces paires nous avons également besoin d'une valeur pour la demande entre l'origine et la destination. La demande pour les paires O-D influe sur la valeur de la fonction objectif mais ne change pas la position de ses points de brisure. En effet, quelle que soit la demande, l'ensemble des plus courts chemins issus d'une origine reste le même. Ceci implique que les points de brisure de la fonction objectif ne dépendent que des coûts sur les arcs et non de la valeur de la demande. Pour cette raison nous avons fixé celle-ci, pour nos tests, à 1 pour toutes les paires O-D des deux réseaux considérés.

3.8.1 Le réseau de Sioux Falls

Ce réseau comporte 24 sommets et 76 arcs (voir [11]). Les valeurs de F ont été choisies sur l'intervalle $[2, 20]$ et celles de G sur l'intervalle $[0, 20]$. Chaque sommet est à la fois une origine et une destination, de sorte que le réseau possède $24 \times 23 = 552$ paires O-D. La figure 3.16 illustre l'approximation U avec 20 points, qui est une enveloppe supérieure de la fonction objectif.

La figure 3.17 donne la valeur de la borne sur l'erreur obtenue avec les méthodes DYN et UNI. En abscisse se trouve le nombre de points d'évaluation. En ordonnée se trouve l'erreur, exprimée cette fois comme pourcentage non pas de la valeur exacte de l'intégrale, qui est inconnue, mais de l'intégrale de l'approximation L : l'erreur est obtenue en divisant $\int_0^1 (U - L)$ par $\int_0^1 L$ puis en multipliant par 100. Puisque L est une sous-évaluation de la fonction objectif, ce pourcentage donne une borne supérieure sur l'erreur relative, car

$$\frac{\int_0^1 (U - L)}{\int_0^1 f} \leq \frac{\int_0^1 (U - L)}{\int_0^1 L}.$$

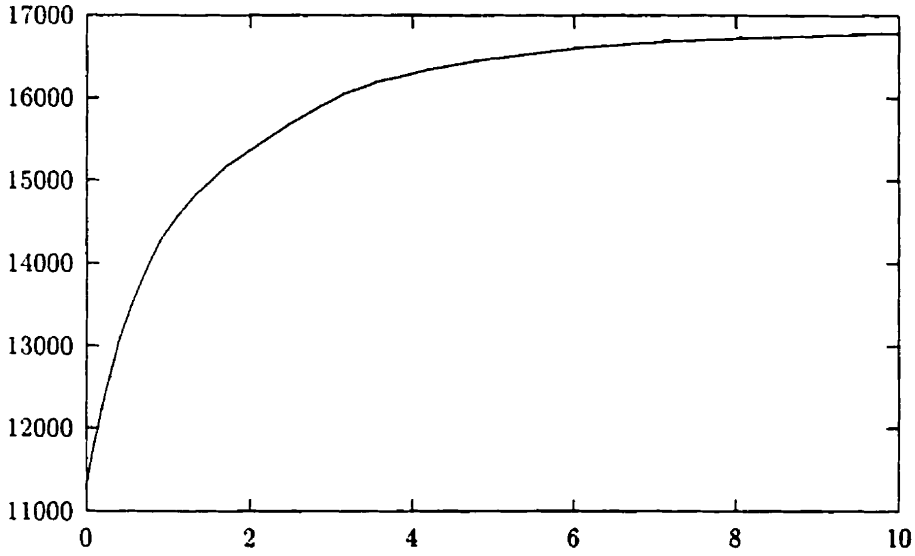


Figure 3.16 – Réseau de Sioux Falls : enveloppe supérieure

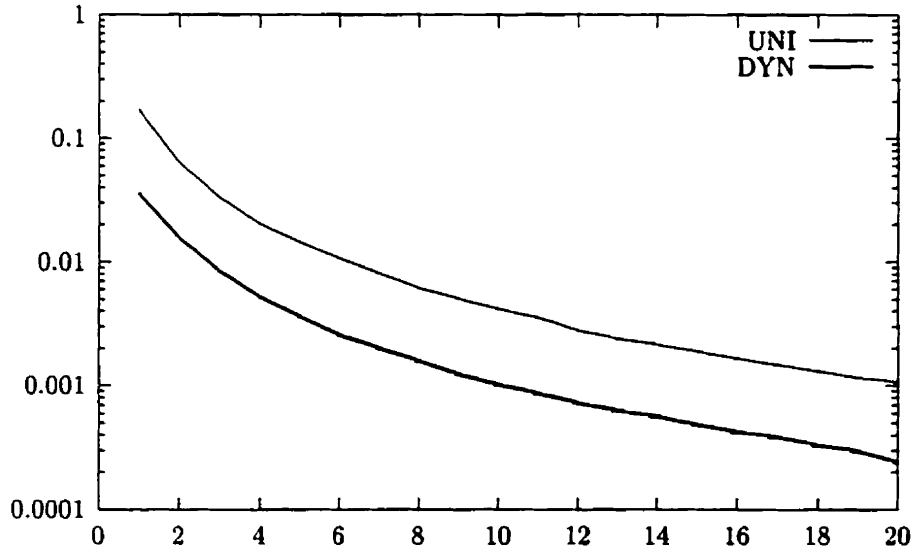


Figure 3.17 – Réseau de Sioux Falls : erreur maximale

3.8.2 Le réseau de Montréal

Ce réseau comporte 9987 sommets et 19304 arcs. Les valeurs de F ont été choisies sur l'intervalle $[0.002, 0.02]$ et celles de G sur l'intervalle $[0, 0.02]$. Il y a 145 origines et pour chacune, le nombre de destinations a été choisi aléatoirement entre 1 et 150. Au total, le réseau possède 10868 paires O-D. La figure 3.18 illustre l'enveloppe supérieure U obtenue avec 20 points d'évaluation.

La figure 3.19 donne la valeur de la borne sur l'erreur obtenue avec les méthodes DYN et UNI. Les valeurs des axes du graphique sont les même que pour le réseau de la section précédente.

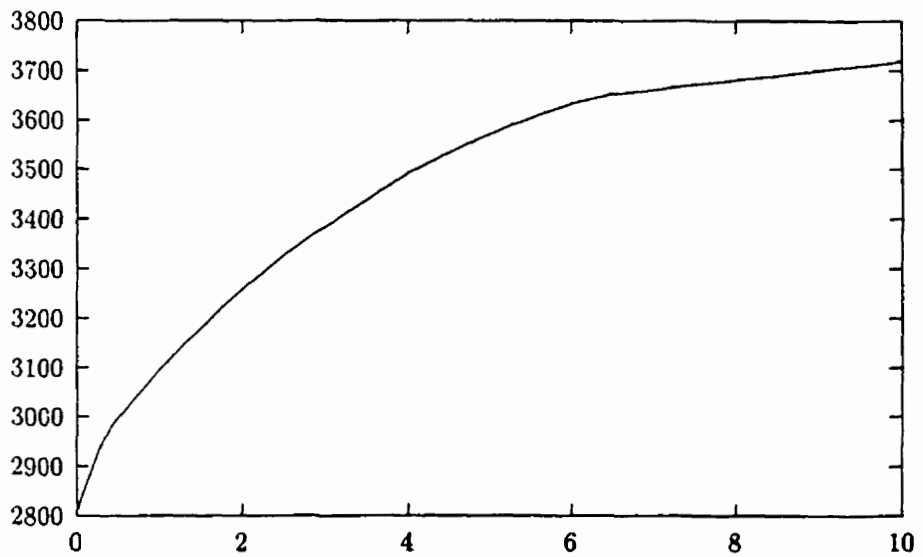


Figure 3.18 – Réseau de Montréal : enveloppe supérieure

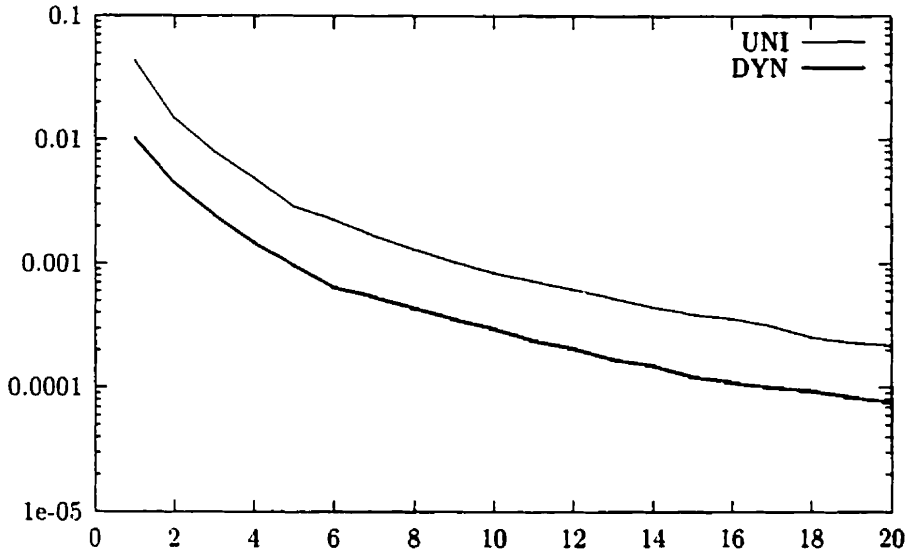


Figure 3.19 – Réseau de Montréal : erreur maximale

CHAPITRE 4

CONCLUSION ET EXTENSIONS

4.1 Conclusion

Dans ce mémoire, nous avons développé une nouvelles méthode adaptative pour l'approximation de fonctions concaves croissantes. Basée sur un résultat théorique, cette méthode est optimale, sous les hypothèses considérées. En pratique, ceci permet la construction d'une procédure itérative simple pour l'approximation de telles fonctions, et nous avons vérifié par des expériences numériques qu'elle donne de bons résultats lorsque comparée à une approximation uniforme.

Nous avons également proposé une nouvelle version de l'algorithme du sandwich, qui diffère par la façon de choisir les points de subdivision de celles que l'on retrouve dans la littérature, et qui utilise la formule obtenue au chapitre 2. Ce faisant, nous avons répondu, pour la norme $\mathcal{L}^1$, à la question posée dans [19] au sujet d'un algorithme du sandwich optimal. En pratique, nos tests ont montré qu'il s'agit d'une procédure efficace.

D'un côté plus pratique, nous avons proposé une application de ces méthodes au problème d'équilibre bicritère.

Enfin, tant du point de vue théorique que pratique, il y encore des questions intéressantes à étudier. Nous en mentionnons deux.

4.2 Extensions

4.2.1 Généralisation à une fonction concave quelconque

Le théorème 1 du chapitre 2 a été démontré pour une fonction concave croissante sur l'intervalle [0,1]. Il est possible, avec quelques modifications, de formuler le problème pour une fonction concave mais pas nécessairement croissante. Considérons une fonction f concave et normalisée (c'est-à-dire, comme auparavant que $f(0) = 0$ et $f(1) = 1$). La situation est très semblable à celle de la figure 2.7 du chapitre 2 et est illustrée à la figure 4.1. Sur l'intervalle $[0, x]$ on a de nouveau une fonction concave, que l'on peut normaliser au moyen d'un changement de variable. Pour obtenir une relation de récurrence analogue à 2.2, nous choisissons cette fois-ci une stratégie d'évaluation *de droite à gauche*. La raison pour le choix de cette stratégie est que, contrairement au cas précédent, on n'a pas nécessairement $v \leq 1$ comme nous l'avions supposé lors de la définition de la transformation T du chapitre 2 (section 2.1). Celle-ci ne s'applique donc pas sur l'intervalle $[x, 1]$. Par contre, on peut utiliser T sur $[0, x]$. Définissons la fonction $\tilde{\mathcal{E}}_n$ comme suit

$\tilde{\mathcal{E}}_n(a, b)$ est la valeur minimum de l'erreur maximale, sachant que l'on peut effectuer n évaluations de f selon la stratégie droite à gauche. On a

$$\tilde{\mathcal{E}}_n(a,b) = \min_x \max_v \max_\mu \left\{ xv\tilde{\mathcal{E}}_{n-1}\left(\frac{v}{x}a, \frac{v}{x}\mu\right) + \frac{(\mu - \mu x - 1 - v)(bx + 1 - b - v)}{\mu - b} \right\}.$$

Les contraintes sur x, v et μ sont identiques à celle de 2.2. Le premier terme entre accolades exprime l'erreur maximale sur $[0, x]$ étant donné qu'il reste $n - 1$ évaluations à faire et le deuxième terme est l'erreur maximale sur $[x, 1]$, c'est-à-dire l'aire du triangle LNK sur la figure 4.1. La relation ci-dessus est très semblable à

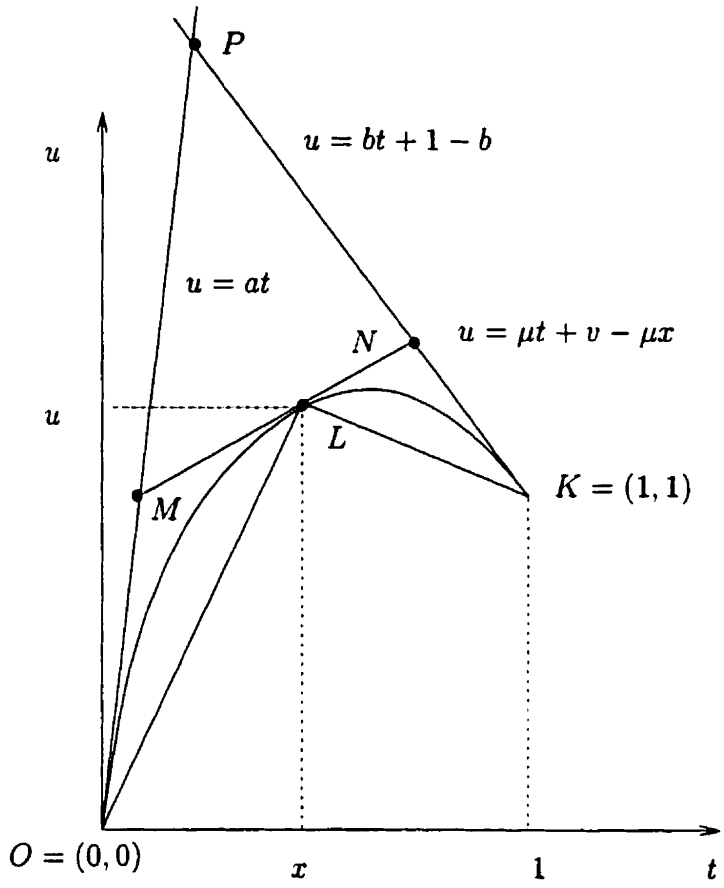


Figure 4.1 – Fonction concave quelconque.

celle du chapitre 2, n'en différant seulement que par l'ordre d'évaluation. Nous croyons donc qu'il est possible de prouver, de façon essentiellement semblable à la preuve du théorème 1, que comme auparavant

$$\bar{\mathcal{E}}_n(a,b) = \frac{(a-1)(1-b)}{2(n+1)^2(a-b)}. \tag{4.1}$$

Pour démontrer ceci, la preuve du théorème 1 devra être modifiée sous deux aspects. D'abord pour l'adapter à l'ordre d'évaluation de droite à gauche et ensuite pour tenir compte du fait que les valeurs de b et μ ne sont plus nécessairement positives. S'il s'avère que l'on peut ainsi démontrer l'égalité 4.1 nous aurons une généralisation

intéressante du théorème 1. Cependant le minimum donné dans la deuxième partie de ce théorème serait différent puisque l'ordre des points n'est plus le même.

4.2.2 Plus court chemin paramétrique avec une distribution non uniforme des usagers

À la section 1.2.1 du chapitre 2, nous avons présenté une application de la méthode DYN à l'approximation de la solution du problème de plus court chemin paramétrique. Ceci nous permet de déterminer le chemin emprunté par les usagers des différentes classes α. Avec cette information, et connaissant la demande entre les origines et les destinations ainsi que la distribution des classes α dans la population, on peut calculer le flot sur chacun des chemins du réseau. L'approximation de ces flots à l'aide de DYN ne tient pas compte de la distribution des classes d'usagers. Plus précisément, celle-ci est supposée uniforme. En effet, le même poids est accordé à l'erreur commise sur les intervalles où la proportion des usagers est faible qu'à celle sur les intervalles où elle est forte. Il serait intéressant de pondérer l'erreur selon la distribution de α de sorte que lors de la minimisation on tienne compte de la répartition des classes dans la population.

Pour ce faire il faudrait calculer l'erreur maximale en la pondérant au moyen de la fonction de densité h. On peut énoncer le problème de minimiser l'erreur maximale et poser la relation de récurrence de la même façon qu'au chapitre 2, à ceci près que pour $n = 0$, la base de la récurrence, on a maintenant

$$\mathcal{E}_0(a,b) = \int_0^1 h(x)(U(x) - L(x))\, dx. \tag{4.2}$$

Cependant, l'introduction de la fonction h complique de beaucoup les calculs : là où l'on devait intégrer une fonction linéaire par morceaux on doit maintenant travailler

avec une fonction plus générale. Par conséquent, il semble peu probable que l'on puisse obtenir une expression explicite pour l'erreur maximale comme celle du théorème 1. Il faudrait alors tenter de calculer numériquement la valeur de $\mathcal{E}_0(a,b)$, ce qui, selon nos expériences, peut se révéler problématique et inefficace. Pour contourner ces difficultés, nous avons envisagé deux méthodes heuristiques pour résoudre le problème de minimisation de l'erreur maximale pondérée par la fonction de densité h.

La première consiste simplement à discrétiser la fonction h, c'est-à-dire l'approximer par une fonction de la forme

$$\tilde{h}(x) = c_i \text{ si } x \in [x_{i-1}, x_i], \quad i = 1, \ldots, M.$$

L'expression de $\mathcal{E}_0$ prend alors la forme

$$\mathcal{E}_0(a,b) = \sum_{i=1}^{M} c_i \int_{x_{i-1}}^{x_i} (U(x) - L(x))\, dx,$$

ce qui semble être plus facile à analyser que l'expression 4.2.

Une autre approche possible est de calculer $\mathcal{E}_0(a,b)$ en assignant un poids à chacun des intervalles $[0,x]$ et $[x,1]$ de la figure 2.7, en fonction de h. Précisément, soit $H(x) = \int_0^x h(t)\, dt$. On pose

$$\begin{aligned} \mathcal{E}_n(a,b) \;=\; & \min_x \max_v \max_\mu \Big\{ xvH(x)\mathcal{E}_0\left(\tfrac{v}{x}a, \tfrac{v}{x}\mu\right) \\ & + (1-x)(1-v)(1-H(x))\mathcal{E}_{n-1}\left(\tfrac{1-v}{1-x}\mu, \tfrac{1-v}{1-x}b\right) \Big\}. \end{aligned}$$

Avec cette formulation on ne change pas l'expression de $\mathcal{E}_0$ et de plus H ne dépend que de la variable x. Il est donc possible que l'on puisse utiliser certains éléments de la preuve du théorème 1 pour calculer le maximum sur v et μ, quitte à recourir à une méthode numérique pour la minimisation sur x.

Comme on le voit, la généralisation au cas où h n'est pas uniforme présente des défis importants. Cependant, tant du point de vue théorique que pratique il serait intéressant de résoudre ce problème. Dans ce but, il pourrait être profitable d'explorer plus à fond les deux approches de solution que nous indiquons.

RÉFÉRENCES

[1] AVRIEL, M. (1976). Nonlinear programming. Analysis and methods, *Prentice-Hall*, Englewood Cliffs, New Jersey.

[2] BELLMAN, R. et DREYFUS, S. (1962). Applied Dynamic Programming, *Princeton University Press*, Princeton.

[3] BRAESS, D. (1968). Über ein Paradoxon der Verkehrsplannung, *Unternehemensforschung* **12**, 256-268.

[4] BURKARD, R.E., HAMACHER, H.W. et ROTE, G. (1991). Sandwich approximation of univariate convex functions with an application to convex separable convex programming, *Naval Research Logistics* **38**, 911-924.

[5] CHVÁTAL, VAŠEK (1980). Linear programming, *W.H. Freeman and Company*, New York.

[6] DeVORE, R. A. (1998). Nonlinear approximation, *Acta Numerica*, 51-150.

[7] FRUHWIRTH, B., BURKARD, R.E. et ROTE, G. (1989). Approximation of convex curves with application to the bicriterial minimum cost flow problem, *European Journal of Operational Research* **42**, 326-338.

[8] GAL, S. et MICCHELLI, A. (1980). Optimal sequential and non-sequential procedures for evaluating a functional, *Applicable Analysis* **10**, 105-120.

[9] GRUBER, P.M. (1992). Aspects of approximation of convex bodies in *Handbook of convex geometry*, Gruber, P.M., Wills, J.M. (eds), *North-Holland*, Amsterdam.

[10] IRELAND, N. J. (1992). Product differentiation and quality, in *The New Industrial Economics*, G. Norman and M. La Manna eds, *Edward Elgar Publishing*, 84-106.

[11] LeBLANC, L.J., MORLOK, E.K., PIERSKALLA, W.P. (1975). An efficient approach to solving the road network equilibrium traffic assignment problem, *Transportation Research* **9**, 309-318.

[12] MARCOTTE, P. (1998). Reformulation of a bicreterion equilibrium model in *Reformulation - Nonsmooth, piecewise smooth, semismooth and smoothing methods*, M. Fukushima and L. Qi eds, *Kluwer*, Dordrecht, 269-292.

[13] MARCOTTE, P. et ZHU, D. L. (1997). Equilibria with infinitely many differentiated classes of customers, in *Complementary and Variational Problems, State of the Art, Proceedings of the 13th International Conference on Complementarity Problems*, Jong-Shi Pang and Michael Ferris eds, *SIAM*, Philadelphia, 234-258.

[14] MARCOTTE, P. (1997). Inéquations variationnelles : motivation, algorithmes de résolution et quelques applications, *Cours donné à Zinal, Suisse, 4-8 mars.*

[15] MARCOTTE, P., NGUYEN S. et TANGUAY, K. (1996). Implementation of an efficient algorithm for the multiclass traffic assignement problem, *Proceedings of the 13th International Symposium on Transportation and Traffic Theory*, Lyon, Jean-Baptiste Lesort ed, *Pergamon*, 217-236.

[16] NOVAK, E. (1996). On the power of adaption, *Journal of Complexity* **12**, 199-237.

[17] NOVAK, E. (1995). Optimal recovery and n-widths for convex classes of functions, *Journal of Approximation Theory* **80**, 390-408.

[18] NOVAK, E. (1995). The adaption problem for nonsymmetric convex sets, *Journal of Approximation Theory* **82**, 123-134.

[19] ROTE, G. (1992). The convergence rate of the sandwich algorithm for approximating convex functions, *Computing* **48**, 337-361.

[20] SONNEVEND, G. (1984). Sequential algorithms of optimal order global error

for the uniform recovery of functions with monotone $(r-1)$ derivatives, *Analysis Mathematica* **10**, 311-335.

[21] SONNEVEND, G. (1983). An optimal sequential algorithm for the uniform approximation of convex functions on $[0,1]^2$, *Applied Mathematics and Optimization* **10**, 127-142.

[22] TANGUAY, K. (1997). Implantation d'un algorithme bicritère d'affectation de traffic, *Mémoire de maîtrise de l'Université de Montréal.*

[23] TRAUB, J.F. et WOŹNIAKOWSKI, H. (1980). A General Theory of Optimal Algorithms, *Academic Press*, New York.

[24] TRAUB, J.F., WASILKOWSKI, G.W. et WOŹNIAKOWSKI, H. (1988). Information-based complexity, *Academic Press*, New York.

[25] YANG, X.Q. et GOH, C.J. (1997). A method for convex curve approximation, *European Journal of Operational Research* **97**, 205-212.